Oracle® Fusion Middleware

Administering Oracle WebCenter Portal

11*g* Release 1 (11.1.1.8.3)

E27738-06

June 2014

Explains how to manage Oracle WebCenter Portal. System administrators will learn how to deploy portals, configure back-end services, monitor performance, implement backup and recovery strategies, and more.

ORACLE

Oracle Fusion Middleware Administering Oracle WebCenter Portal, 11*g* Release 1 (11.1.1.8.3)

E27738-06

Primary Author: Rosie Harvey

Contributing Authors: Sarah Bernau, Michele Cyran, Sue Highmoor, Peter Jacobsen, Shahana Mitra, Ingrid Snedecor, Savita Thakur

Contents

9 Managing Content Repositories

10 Managing Activity Graph

11 Managing Analytics

12 Managing Announcements and Discussions

13 Managing Calendar Events

14 Managing Instant Messaging and Presence

15 Managing Mail

16 Managing People Connections

17 Managing RSS

18 Managing Oracle Secure Enterprise Search in WebCenter Portal

19 Managing Subscriptions and Notifications

20 Managing Worklists

21 Managing Portlet Producers

22 Managing the Pagelet Producer

23 Managing External Applications

24 Managing REST Services

25 Managing Personalization

30 Managing Oracle WebCenter Portal Security

31 Configuring the Identity Store

32 Configuring the Policy and Credential Store

33 Configuring Single Sign-on

34 Configuring Portal Framework Applications for Single Sign-on

35 Configuring SSL

36 Configuring WS-Security

39 Understanding WebCenter Portal Life Cycle

40 Deploying Portals, Templates, Assets, and Extensions

41 Managing WebCenter Portal Backup, Recovery, and Cloning

Part IX Lifecycle: Portal Framework Applications

42 Deploying Portal Framework Applications

43 Administering Portal Framework Applications Using the Administration Console

44 Managing Export, Import, Backup, and Recovery for Portal Framework Applications

49 Managing Security Across Portals

50 Customizing System Pages

51 Managing Business Role Pages

52 Managing Personal Pages

53 Administering Device Settings

E Labeling During WebCenter Portal Lifecycle

F Migrating Wiki Content to WebCenter Portal

G Troubleshooting Oracle WebCenter Portal

Preface

Welcome to Oracle Fusion Middleware Administering Oracle WebCenter Portal!

Audience

This document is intended for system administrators responsible for Oracle WebCenter Portal installations.

This guide assumes that the audience is familiar with the concepts and content described in *Oracle Fusion Middleware Administrator's Guide*.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc`.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info` or visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs` if you are hearing impaired.

Related Documents

For more information, see the following documents in the Oracle Fusion Middleware 11*g* Release 1 (11.1.1.8.3) documentation set:

- *Oracle Fusion Middleware Administrator's Guide*
- *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*
- *Oracle Fusion Middleware Using Oracle WebCenter Portal*
- *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*
- *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

What's New?

The following topics introduce the new and changed features of WebCenter Portal and other significant changes that are described in this guide, and provides pointers to additional information. This book is the new edition of the formerly titled *Administrator's Guide for Oracle WebCenter Portal: Spaces*.

New and Changed Features for 11*g* Release 1 (11.1.1.8.3)

- Support for four new page templates that provide efficiency and performance enhancements over the existing page templates. See "New Page Templates."
- Support for the FrameworkFolders component that provides a scalable, high-performing folder service from Oracle WebCenter Content as an alternative to Folders_g. See "FrameworkFolders Support."

New Page Templates

WebCenter Portal Bundle patch 11.1.1.8.3 introduces support for four new page templates that provide efficiency and performance improvements over the existing page templates. The files included are:

- Four *page templates*: Skyros Side Navigation v2, Skyros Side Navigation (Stretch) v2, Skyros Top Navigation v2, and Skyros Top Navigation (Stretch) v2
- One *skin*: Skyros v2

  This is the preferred skin for the new page templates. Do not attempt to use the new Skyros v2 skin with existing page templates. Likewise, do not attempt to use the new page templates with the existing skins.
- Two *task flows*: Portal Side Navigation and Portal Top Navigation

  These task flows are used by the page templates to implement the navigation, either in a side pane or as tabs along the top of a portal. The styling of the navigation in these task flows relies on the CSS in the skin file available in this patch. If you want to use these task flows in a page template that uses a different skin, be aware that the navigation may not look as expected due to CSS mismatches in the skin. However, if you copy and paste the navigation sections from the new Skyros v2 skin source code into your skin source code, you can achieve the expected results.

To use the new page templates:

1. Locate `18085041/NewAssetBP3.zip` in WebCenter Portal Bundle patch 11.1.1.8.3 (patch 18085041) and extract the contents to a local directory.
2. Log in to WebCenter Portal.

3. Go to the **Shared Assets** page, which is available if you have the permissions of the `Administrator` or `Application Specialist` role.
4. Select the following assets in the left pane, and upload the corresponding files from the saved location on your local file system:

Asset Type	File Names

Page Templates	`pageTemplate_Skyros_Side_Navigation_Stretch_v2.ear`
	`pageTemplate_Skyros_Side_Navigation_v2.ear`
	`pageTemplate_Skyros_Top_Navigation_Stretch_v2.ear`
	`pageTemplate_Skyros_Top_Navigation_v2.ear`

Skins	`skin_Skyros_v2_Skin.ear`
Task Flows	`taskFlow_Portal_Side_Navigation.ear` `taskFlow_Portal_Top_Navigation.ear`

After uploading the page templates, skin, and task flows, perform the following steps to use the new page templates:

1. To set a new default page template for pages in the Home portal and all new portals (when the portal's template does not specify that a particular page template must be used), see Section 43.3.1, "Choosing a Default Page Template."
2. To change the page template used by an individual portal, see the "Changing the Page Template for a Portal" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.
3. To change the preferred skin used by a page template, see the "Setting the Preferred Skin for a Page Template" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

For more information, see the "Working with Page Templates" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal* and the "Developing Page Templates" chapter in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

FrameworkFolders Support

Previously, Oracle WebCenter Portal only supported Folders_g. WebCenter Portal Bundle patch 11.1.1.8.3 enables *new* installations of Oracle WebCenter Portal to be integrated with FrameworkFolders. Existing installations of Oracle WebCenter Portal patched to release 11*g* Release 1 (11.1.1.8.3) *must* continue to use Folders_g.

For information about the criteria that must be met for enabling FrameworkFolders, see the "Preparing Oracle WebCenter Portal for FrameworkFolders Support" section in the *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*. For information about the Folders_g and FrameworkFolders directory structure and the steps for enabling FrameworkFolders, see Section 9.2.3.1, "Enabling Mandatory Components."

New and Changed Features for 11*g* Release 1 (11.1.1.8.0)

WebCenter Portal 11*g* Release 1 (11.1.1.8.0) included the following new and changed features:

- Terminology changes:

Prior Releases	Current Release
WebCenter Portal: Spaces	WebCenter Portal
space	portal
space template	portal template
resource	asset

- End-User Experience
  - Updated profile user interface that includes improved organization of profile information, click to edit, and clear profile photo functionality. See the "Managing Your Profile" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*. Additional documentation for the rich user profile is referenced under Portal Builder, Administration, and Development Environment.
  - Improved search experience (supported with Oracle SES 11.2.2.2) that includes faceted search and document thumbnails. See the "Searching for Information" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.
- Portal Builder
  - Simplified portal creation that includes in-place page creation. See the "Creating and Building a New Portal" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.
  - Redesigned portal edit and administration user interface (Portal Builder) that consolidates tasks into fewer steps. See the "Editing a Portal" and "Administering a Portal" chapters in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.
  - Simplified page creation and editing: Web (for editing) and Data (for managing) views, inline resource catalog (with support for component drag-and-drop onto a page), and Select view. See the "Working with Portal Pages" part in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.
  - Automatic update of portal navigation as new pages are created. See the "Creating a Page or Subpage in an Existing Portal" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.
  - "Lazy provisioning" of tools—WebCenter Portal configures the back-end server at first use of a tool rather than at portal creation to speed the successful creation of a new portal. See the "About Creating a New Portal" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.
  - Hierarchical page support (subpages). See the "Creating Pages or Subpages in a Portal" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.
  - Updated profile user interface that includes improved organization of profile information, click to edit, and clear profile photo functionality; new component properties for improved control of people connections and activity graph components. See the "Adding Activity Graphs and Recommendations to a Portal," "Adding Connections to a Portal," and "Adding Profiles to a Portal" chapters in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*. Additional documentation for the rich user profile is

referenced under End-User Experience, Administration, and Development Environment.

  - Device Settings that control how your portal pages render on different devices, such as smart phones, tablets, and desktop browsers. Page variants can be created to target and optimally render a portal on specific groups of devices like iOS phones, iOS tablets, and others. See the "Administering Device Settings in a Portal" section, the "Managing Device Groups for a Portal" chapter, and the "Creating a Page Variant for a Device Group" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*. Additional documentation for mobile support is referenced under Administration and Development Environment.
  - Responsive Content Presenter templates that provide an example of how you can use Content Presenter and CSS3 media queries to produce a responsive layout that adjusts to the width of the browser (for example, on smart phones, tablets, and desktop browsers). See the "Using Responsive Templates" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*. Additional documentation for mobile support is referenced under Development Environment.

- Administration
  - Simplified WebCenter Portal administration that includes a power user oriented experience with familiar concepts for legacy WebCenter Portal customers. See Part XI, "Managing Portals in Portal Builder Administration".
  - New profile configuration settings that include properties to specify whether to show the new or legacy profile user interface and to specify profile synchronization settings. See Chapter 16, "Managing People Connections." Additional documentation for the rich user profile is referenced under End-User Experience, Portal Builder, and Development Environment.
  - Device Settings that control how your portal pages render on different devices, such as smart phones, tablets, and desktop browsers. Page variants can be created to target and optimally render a portal on specific groups of devices like iOS phones, iOS tablets, and others. See Section 40.4, "Deploying Devices and Device Groups," Section 50.2.2, "Creating a Page Variant of a System Page for Device Groups," and Chapter 53, "Administering Device Settings." Additional documentation for mobile support is referenced under Portal Builder and Development Environment.
  - Impersonation, which allows a privileged user to impersonate another user for the purposes of verifying the other user's experience in WebCenter Portal and troubleshooting unexpected results. See Chapter 38, "Managing Impersonation."
  - Improved portal lifecycle tools that enable export/import and backup/recovery of one or more portals with minimal downtime. See Chapter 40, "Deploying Portals, Templates, Assets, and Extensions" and Chapter 41, "Managing WebCenter Portal Backup, Recovery, and Cloning."
  - Integrated Oracle WebCenter Portal's Pagelet Producer user interface within WebCenter Portal's administrative user interface to make system administrators aware of the existence of Pagelet Producer pagelets and to allow them to make these pagelets available to end users. Integrating the UIs also provides Pagelet Producer developers to easily navigate from WebCenter Portal where they see the pagelets to the Pagelet Producer Administration UI so they can create new or edit existing pagelets. See Chapter 22, "Managing the Pagelet Producer."

- New page performance analyzer that shows you how long individual components take to display on a portal page, as well as the overall time taken to display a page. This new tool is useful to developers who are performing first level performance analysis, customers who build their own pages, and any user who customizes pages in WebCenter Portal. See Section G.4.4, "How to Identify Slow Page Components."

- Development Environment
  - Updated profile user interface that includes improved organization of profile information, click to edit, and clear profile photo functionality; new component properties for improved control of people connections and activity graph components. See the "Introducing the People Connections Service," "People Connections Task Flow Binding Parameters," and "Integrating the Activity Graph" chapters in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*. Additional documentation for the rich user profile is referenced under End-User Experience, Portal Builder, and Administration.
  - Developers can use Expression Language (EL) to retrieve information about Device Settings. Device Settings control exactly how your portal pages render on different devices including smart phones, tablets, and desktop browsers. See the "EL Expressions Related to Device Settings" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*. Additional documentation for mobile support is referenced under Portal Builder and Administration.
  - Responsive Content Presenter templates provide an example of how you can use Content Presenter and CSS3 media queries to produce a responsive layout that adjusts to the width of the browser (for example, on phones, tablets, or personal computers). See the "Using Responsive Templates" section and the "Extending Responsive Templates" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*. Additional documentation for mobile support is referenced under Portal Builder.
  - Simplified custom shared library development and deployment. WebCenter Portal provides a new JDeveloper template that enables you to build custom components, such as task flows, data controls, and managed beans and deploy them in shared libraries directly to the WebCenter Portal server. See the "Developing Components for WebCenter Portal Using JDeveloper" chapter in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

- Restructured documentation library according to personas and their roles in WebCenter Portal:
  - *Oracle Fusion Middleware Using Oracle WebCenter Portal* covers information needed by a *knowledge worker* who typically uses WebCenter Portal to contribute and review content, participate in social interactions, and leverage the Home portal to manage her own documents and profile.
  - *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal* covers information needed by an *application specialist* who works in Portal Builder to create and administer portals, their structure (hierarchy of pages, navigation, security), and their content (components on a page, layout, behavior, and so on).
  - *Oracle Fusion Middleware Administering Oracle WebCenter Portal* (this guide) covers information needed by a *system administrator* who fields requests from

IT employees and business users to set up new machines; clone or back up existing applications systems and databases; install patches, packages, and applications; and perform other administration-related tasks.

– *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper* covers information needed by a *developer* who primarily works with JDeveloper to provide support for both portals and WebCenter Portal Framework applications.

For more information, see "Who's Who."

Who's Who

Throughout this guide, we provide examples that illustrate some of the ways you can use WebCenter Portal. The tasks in the guide are targeted to one or more personas, assigned specific portal roles. This chapter introduces you to these personas and describes the ways in which they might interact with WebCenter Portal. These example personas are for illustrative purposes to help you identify the different skill sets required to use the range of tools offered by WebCenter Portal.

The personas described here have the default roles provided out-of-the-box with WebCenter Portal. These roles are each given a unique set of permissions appropriate for the work that each persona will typically do. Note that you can modify these default roles or configure new roles to meet the unique needs of your organization.

The people who interact with WebCenter Portal typically work together as a team to coordinate their tasks in one of the following user roles:

- Knowledge Worker
- Application Specialist
- Web Developer
- Developer
- System Administrator

Knowledge Worker

Karen is a *knowledge worker* who typically uses WebCenter Portal to contribute and review content, participate in social interactions, and leverage the Home portal to manage her own documents and profile.

At the application level, Karen has permissions such as those granted to the default `Authenticated-User` role, which may be customized for the specific needs of the organization. At the portal level, the portal `Moderator` will likely assign Karen the `Viewer` or `Participant` role, or a custom role that offers a similar set of permissions.

For more information about roles and permissions, see the "About Roles and Permissions for a Portal" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Knowledge Worker Tasks in WebCenter Portal

Tasks that are typical of a knowledge worker like Karen include:

- Connecting to and collaborating with other WebCenter Portal users by sharing information, files, and links; and by interacting through instant messaging, mail, message boards, discussions, wikis, and blogs
- Uploading, sharing, and managing documents stored in Content Server
- Joining a team or project portal
- Keeping up with changes in WebCenter Portal by receiving notifications when content is updated, exploring recommendations from other users, viewing the activities of the portals she is a member of and users she's connected to, viewing announcements, taking polls, and monitoring WebCenter Portal RSS feeds
- Staying organized through the use of favorites, notes, calendars, lists, links to portal objects, and tags
- Viewing and responding to worklist items

As Karen becomes more familiar with the functionality available in WebCenter Portal, she may begin to perform more advanced tasks, such as creating portals. As a more advanced knowledge worker, her role may evolve to overlap with application specialist tasks.

Information targeted for knowledge workers like Karen is in *Oracle Fusion Middleware Using Oracle WebCenter Portal*. Advanced tasks that overlap with those of an application specialist are covered in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Application Specialist

Ari is an *application specialist* who works in Portal Builder to create and administer portals, their structure (hierarchy of pages, navigation, security), and their content (components on a page, layout, behavior, and so on). In a typical project, Ari coordinates the efforts of Karen (knowledge worker), Wendy (web developer), and Dave (developer).

At the application level, Ari has permissions such as those granted to the default `Application Specialist` role, which may be customized for the specific needs of the organization. In a portal that Ari creates, he performs actions available to the `Moderator` role to manage the portal.

For more information about roles and permissions, see the "About Roles and Permissions for a Portal" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Application Specialist Tasks in WebCenter Portal

Tasks that are typical of an application specialist like Ari include:

- Planning and creating new portals
- Editing and administering the portals he owns
- Creating and building portal pages using the page editor (Composer) and the resource catalog to add and configure page components
- Creating and managing portal assets, tools, and services
- Managing shared assets and portal templates across all portals

Information targeted for application specialists like Ari is in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*. To work with his personal view of the Home portal, Ari will also refer to *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

Web Developer

Wendy is a *web developer* who focuses on delivering a consistent, branded look and feel to all portals. Wendy provides graphics designs and HTML markup from which Ari (application specialist in Portal Builder) or Dave (developer in JDeveloper) can create content or page style templates, skins, and so on. Once these assets are created, Ari can leverage them to create portal pages. Wendy typically does not interact with WebCenter Portal directly.

Web Developer Tasks in WebCenter Portal

Tasks that are typical of a web developer like Wendy include:

- Developing a corporate portal look and feel
- Designing new portal page templates

Information targeted for web developers like Wendy is in the "Creating a Look and Feel for Portals" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Developer

Dave is a *developer* who provides support for both portals and WebCenter Portal Framework applications:

- **Portals (Portal Builder)**

  Dave is primarily responsible for developing components (such as task flows, page templates, and content templates), which are published and leveraged by Ari (the application specialist). Dave primarily works with JDeveloper and leverages the WebCenter Spaces Extension/WebCenter Portal Service Extension projects.

- **Framework Applications**

  Dave primarily works with JDeveloper to develop WebCenter Portal Framework applications. Once he has developed the application, he can package it as an EAR file and deploy it on the application server. In a typical environment, Dave would have JDeveloper configured with a SCM system and be working within a team with automated build and deploy processes.

Developer Tasks

Tasks that are typical of a developer like Dave include:

- Building and maintaining WebCenter Portal Framework applications
- Developing custom assets, like page templates and navigation components for portals in WebCenter Portal
- Developing Java portlets
- Developing and deploying task flows, managed beans, and other custom components
- Developing custom personalization components
- Maintaining the source control system
- Maintaining a build system

Information targeted for developers like Dave is in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

System Administrator

Syed is a *system administrator* who fields requests from IT employees and business users to set up new machines; clone or back up existing applications systems and databases; install patches, packages, and applications; and perform other administration-related tasks. As the system administrator, Syed works with other tools such as Fusion Middleware Control and command line tools. He leverages Enterprise Manager to configure portal settings, and also configures integrations such as WebCenter Content and other Fusion Middleware products and Oracle applications.

In WebCenter Portal's Portal Builder, he has permissions such as those granted to the default `Administrator` role, which provides exclusive access to administer and set global options for all portals (including the Home portal).

For more information about application level roles and permissions, see the "About Application Roles and Permissions" section in *Oracle Fusion Middleware Administering Oracle WebCenter Portal*.

System Administrator Tasks

Tasks that are typical of a system administrator like Syed include:

- Uses Portal Builder administration to administer all portals (including import and export of portals) and security site-wide
- Uses Portal Builder administration to manage site-wide system pages, business role pages, and personal pages
- Uses Portal Framework application administration console to manage application-wide preferences, manage users and roles, manage assets, configure the content repository, create polls, register producers and external applications
- Leads security, taxonomy, metadata, workflow, governance
- Uses the management console for administrative functions
- Executes command line utilities for administrative functions
- Installs and configures production versions of developers' efforts
- Performs patching of the production versions and the operating system
- Creates clones and backups of the production versions
- Performs restores of production versions
- Monitors the operating system for issues with the production version
- Deploys and redeploys applications

Information targeted for system administrators like Syed is in *Oracle Fusion Middleware Administering Oracle WebCenter Portal* and *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

Part I

Introduction to Oracle WebCenter Portal

This part of *Oracle Fusion Middleware Administering Oracle WebCenter Portal* introduces you to Oracle WebCenter Portal and its administration tools.

Part I contains the following chapter:

- Chapter 1, "Introduction to Administering Oracle WebCenter Portal"

1

Introduction to Administering Oracle WebCenter Portal

This chapter provides a high-level overview of Oracle WebCenter Portal and its administrative tools.

This chapter includes the following topics:

Oracle Fusion Middleware Administering Oracle WebCenter Portal is written specifically for Oracle WebLogic Server—the primary platform for Oracle Fusion Middleware software components such as Oracle WebCenter Portal. If you using a third-party application server provided by a vendor other than Oracle, such as IBM's WebSphere, refer to the *Oracle Fusion Middleware Third-Party Application Server Guide*.

1.1 Introducing Oracle WebCenter Portal

Welcome to Oracle WebCenter Portal!

Oracle WebCenter Portal is an integrated set of components with which you can create social applications, enterprise portals, collaborative communities, and composite applications, built on a standards-based, service-oriented architecture. Oracle WebCenter Portal combines dynamic user interface technologies with which to develop rich internet applications, the flexibility and power of an integrated, multi-channel portal framework, and a set of tools and services that provide content,

collaboration, presence and social networking capabilities. Based on these components, Oracle WebCenter Portal also provides an out-of-the-box enterprise-ready customizable application called *WebCenter Portal*, with a configurable work environment that enables individuals and groups to work and collaborate more effectively.

Oracle WebCenter Portal provides an open and extensible solution that allows users to interact directly with tools like instant messaging, documents, content management, discussion forums, wikis, blogs, and tagging directly from within the context of a portal or an application. These tools and services empower end users and IT to build and deploy next-generation collaborative applications and portals.

This section describes Oracle WebCenter Portal components and architecture in the following topics:

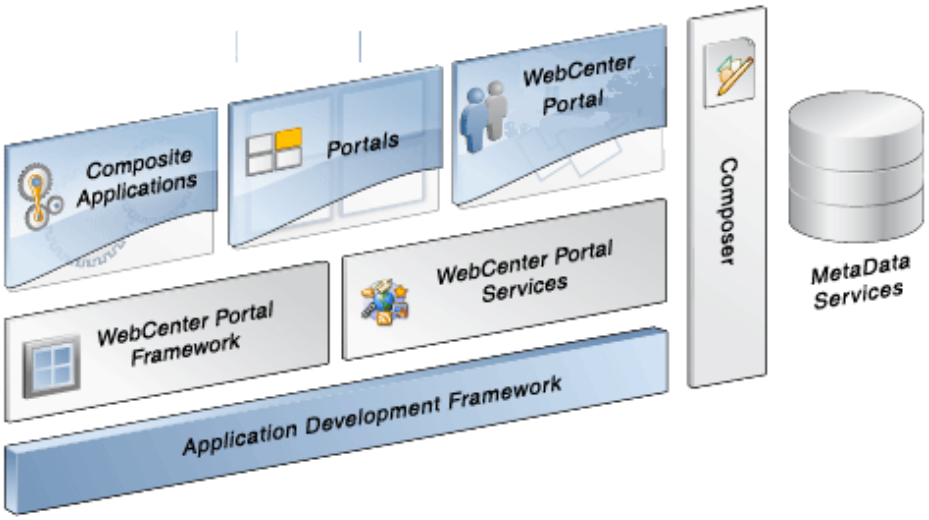
- Section 1.2, "Oracle WebCenter Portal Architecture"
- Section 1.3, "Oracle WebCenter Portal Topology"
- Section 1.4, "WebCenter Portal"
- Section 1.5, "Portal Framework Applications"

1.2 Oracle WebCenter Portal Architecture

Oracle WebCenter Portal comprises the following components (shown in Figure 1–1):

- WebCenter Portal Framework
- Application Development Framework
- WebCenter Portal
- Tools and Services
- Composer
- Discussion Server
- Analytics
- Activity Graph
- Personalization Server
- Portals
- Composite Applications

Figure 1–1 Oracle WebCenter Portal Architecture



1.2.1 WebCenter Portal Framework

Injects portal capabilities into ADF, including:

- Run-time application customization (you can make in-place changes to WebCenter Portal and Portal Framework applications using Composer without re-deploying the application)
- Support for JSR-168 and JSR-286 standards-based WSRP portlets, and PDK-Java portlets
- Content integration through JCR (JSR170), to content repositories such as Oracle WebCenter Content Server and Oracle Portal
- Oracle JSF Portlet Bridge, which lets you expose JSF pages and Oracle ADF task flows as standards-based portlets

1.2.2 Application Development Framework

The Oracle Application Development Framework (ADF) is a productivity layer that sits on top of JSF and provides:

- Unified access to back ends such as databases, web services, XML, CSV, and BPEL
- Data binding (JSR 227) connecting the user interface with back-end data controls
- Over 100 data-aware JSF view components
- Native component model that includes task flows
- Fine grained JAAS security model

1.2.3 Composer

Composer provides:

- Ability to perform run-time application and user customization in-place in your browser
- A rich, intuitive user experience where you can:

  - Browse and add resources, such as task flows and portlets, to pages
  - Re-arrange page layout
  - Set page and component properties
  - Contextually wire components

1.2.4 WebCenter Portal

Out-of-the-box application built using JSF, Oracle ADF, WebCenter Portal Framework, Composer, and tools and services.

WebCenter Portal provides:

- A browser-based platform for creating enterprise portals, multiple sites and communities
- A Home portal for each user, providing a private work area for storing personal content, viewing and responding to business process assignments, emailing, and so on
- Threaded discussions, blogs, wikis, worklists, announcements, RSS, recent activities, search, and more

1.2.5 Tools and Services

Table 1–1 lists the tools and services available in WebCenter Portal and Portal Framework applications.

Table 1–1 WebCenter Portal Tools and Services

Services A Through N	Services P Through W
Activity Graph	Notes[1]
Analytics	People Connections
Announcements	Personalization
Discussions	Polls
Documents (includes Wikis and Blogs)	RSS[2]
Events[1]	Recent Activities
Instant Messaging and Presence (IMP)	Search
Links	Tags
Lists[1]	Worklist
Mail	

[1] WebCenter Portal only.

[2] RSS news feeds are only available in WebCenter Portal. The RSS Viewer task flow is available in WebCenter Portal and other Portal Framework applications.

WebCenter Portal's tools and services provide:

- Seamless integration with enterprise-level services
- Thin adapter layer to abstract back-end services. For example:
  - Content adapters: Content Server and Oracle Portal

  - Presence adapters: Microsoft Live Communications Server, Microsoft Office Communications Server, and Microsoft Lync
- Back-end systems represented by a unified connection architecture
- User interface to services presented through rich task flow components

1.2.6 Discussion Server

A discussion server is provided with Oracle WebCenter Portal so you can integrate discussion forums and announcements into your applications.

1.2.7 Analytics

WebCenter Portal's analytics capability enables users to view various user activity reports, for example:

- Login data
- Page views
- Portlet views
- Document views
- Search metrics
- Page response data
- Portal usage

1.2.8 Activity Graph

Activity graph enables users to analyze various statistics collected by WebCenter Portal analytics. Various similarity scores for objects and users are collected, and used to give recommendations. The scores are stored in an activity graph database.

1.2.9 Personalization Server

WebCenter Portal's personalization server enables you to deliver application content to targeted users based on selected criteria.

1.2.10 Portals

Portals provide a common interface (a Web page) to a personalized, single point of interaction with web-based applications and information relevant to individual users or class of users. For information about creating portals using the WebCenter Portal Framework application template in JDeveloper, see *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

Alternatively, create portals using the out-of-the-box application WebCenter Portal. For details, see *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

1.2.11 Composite Applications

A composite application is an assembly of services, service components, wires, and references designed and deployed as a single application. For more information about composite applications, see *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite*.

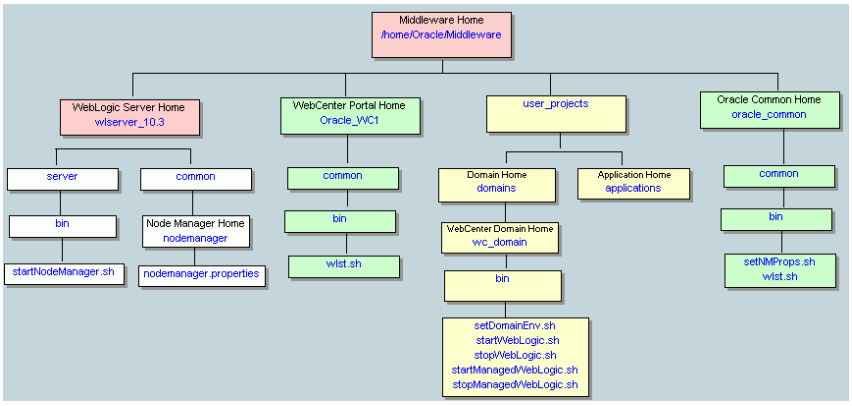
1.3 Oracle WebCenter Portal Topology

This section describes Oracle WebCenter Portal topology and configuration in the following topics:

1.3.1 Oracle WebCenter Portal Topology Out-of-the-Box

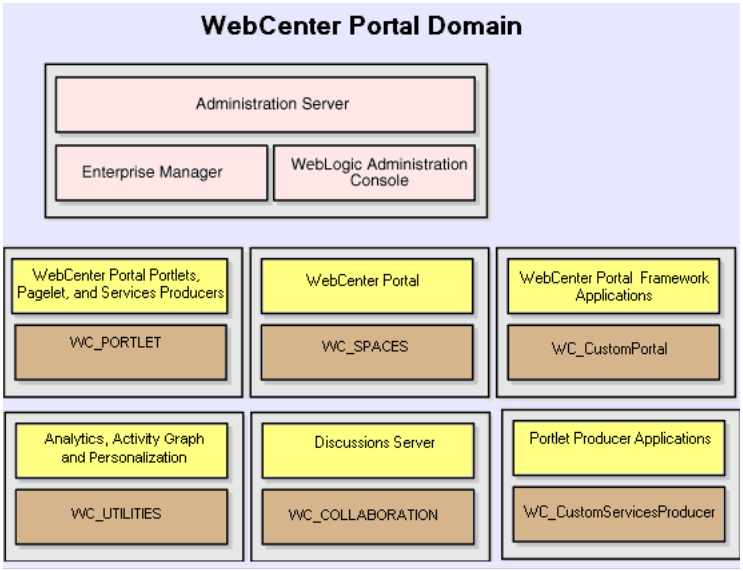
Oracle WebCenter Portal installation creates a **WebCenter Portal Oracle Home** (*WCP_ORACLE_HOME*) under the Oracle Middleware Home directory and an **Oracle Common Home** directory (*ORACLE_COMMON_HOME*), which contains WebCenter Portal binaries and supporting files (Figure 1–2).

Figure 1–2 Directory Structure of an Oracle WebCenter Portal Installation



The installation also creates a WebCenter Portal domain (`base_domain`), containing the administration server and several managed servers to host various WebCenter Portal components. In Figure 1–3, applications are shown in yellow, while the managed servers they run on are shown in brown.

Figure 1–3 Oracle WebCenter Portal Topology Out-of-the-Box



Out-of-the-box managed servers host the following Oracle WebCenter Portal components:

- WC_Spaces - Hosts WebCenter Portal, Oracle's out-of-the-box portal application
- WC_Portlet - Hosts out-of-the-box portlets, the pagelet producer, and the WebCenter Portal services producer
- WC_Collaboration - Hosts the discussions server and any additional services that you choose to integrate
- WC_Utilities - Hosts activity graph, analytics, and personalization services

An optional fifth managed server (an applications server) can be used to run applications built by developers using the WebCenter Portal Framework application template in JDeveloper—such applications are referred to as *Portal Framework applications*. When you create additional managed servers, they are provisioned with the appropriate libraries to enable them to draw upon the same external resources as WebCenter Portal. For more information about managed servers, see the "Understanding Oracle Fusion Middleware Concepts" section in *Oracle Fusion Middleware Administrator's Guide*.

1.3.2 Oracle WebCenter Portal Managed Servers

During Oracle WebCenter Portal installation, the managed servers are provisioned with system libraries and Oracle ADF libraries. Table 1–2 lists the managed servers and the applications that run on them.

Table 1–2 Oracle WebCenter Portal Managed Servers and Applications

Managed Server	Installed Applications	Application Name
WC_Spaces	WebCenter Portal	webcenter
	WebCenter Portal online help	webcenter-help
WC_Portlet	OmniPortlet and Web clipping	portalTools
	WSRP tools	wsrp-tools
	Pagelet producer	pagelet-producer
	Services producer	services-producer
WC_Collaboration	Discussions Server	owc_discussions
WC_Utilities	Analytics collector	analytics-collector
	Activity graph engines	activitygraph-engines
	Personalization services	wcps-services

1.3.3 Oracle WebCenter Portal Startup Order

When a managed server starts up, applications and libraries are started in the following order:

1. Oracle system libraries, known as the JRF libraries.
2. Oracle ADF libraries.
3. Instrumentation applications, such as Oracle DMS, and the Oracle Web Services Manager (`wsm-pm`) application.
4. Oracle WebCenter Portal applications shown in Table 1–2.

The startup order is also the order of dependency. If a dependent component does not deploy successfully, a later component may not function correctly.

Application startup is not dependent on the availability of external services such as the discussions server, or other back-end servers. For details, see Section 1.3.4, "Oracle WebCenter Portal Dependencies."

1.3.4 Oracle WebCenter Portal Dependencies

WebCenter Portal and Portal Framework applications use several external servers, tools, and services (Table 1–3). The Configuration column lists the type of information provided to WebCenter Portal to configure or initialize the connection. The Access column lists the protocol used in run-time access of the service.

Table 1–3 Dependent Resources - Access Types

External Server, Tool or Service	Configuration	Access
Analytics	UDP access to the analytics collector	UDP
Activity graph	HTTP access to activity graph administration	HTTP
Discussions server	HTTP access to discussions server administration	SOAP/HTTP
Content Server (Documents)	Socket connection to the Administration Server. HTTP access is required only if the Content Server must be accessed outside WebCenter Portal.	JCR 1.0 over socket or HTTP

Table 1–3 (Cont.) Dependent Resources - Access Types

External Server, Tool or Service	Configuration	Access
Instant messaging and presence server	HTTP access to instant messaging and presence server administration	SOAP/HTTP
Mail server	IMAP/SMTP server	IMAP/SMTP
Personal events server	HTTP access to calendar services	SOAP/HTTP
Personalization server	JDBC access to the personalization server	JDBC REST
Portlets	HTTP location of provider WSDLs	SOAP/HTTP
Search server	HTTP access to search server	HTTP
Worklist	HTTP access to BPEL server	SOAP/HTTP
MDS and schemas	JDBC	JDBC

Server/service unavailability does not prevent WebCenter Portal or Portal Framework applications from starting up, although errors may display while the application is running. The only exception is the Oracle Metadata Services Repository (MDS), as WebCenter Portal and Portal Framework applications do not work without it.

WebCenter Portal partially works without a repository but only if it is a different physical database from the MDS repository. The WebCenter Portal repository stores information for several services, including events, links, lists, people connections, polls, and tags, and these services do not work if the repository is not available.

1.3.5 Oracle WebCenter Portal Configuration Considerations

The main configuration files for WebCenter Portal and Portal Framework applications are listed and described in Table 1–4. Both these files are supplied within the application deployment .EAR file.

Table 1–4 WebCenter Portal Configuration Files

Artifact	Purpose
`adf-config.xml`	Stores basic configuration for Application Development Framework (ADF) and application settings, such as which discussions server or mail server WebCenter Portal or the Portal Framework application is currently using.
`connections.xml`	Stores basic configuration for connections to external services.

WebCenter Portal, Portal Framework applications, and portlet producers all use the Oracle Metadata Services (MDS) repository to store their configuration data; both access the MDS repository as a JDBC data source within the Oracle WebLogic framework.

The MDS repository stores post deployment configuration changes for WebCenter Portal, Portal Framework applications, and portlet producers as application customizations. MDS uses the original deployed versions of `adf-config.xml` and `connections.xml` as base documents and stores all subsequent application customizations separately into MDS using a single customization layer.

When a WebCenter Portal or Portal Framework application starts up, application customizations stored in MDS are applied to the appropriate base documents and the

application uses the merged documents (base documents with customizations) as the final set of configuration properties.

For applications that are deployed to a server cluster, all members of a cluster read from the same location in the MDS repository.

Typically, there is no need for administrators to examine or manually change the content of base documents (or MDS customization data) for files such as `adf-config.xml` and `connections.xml`, as Oracle provides several administration tools for post deployment configuration. If you must locate the base documents or review the information in MDS, read Appendix A, "Oracle WebCenter Portal Configuration."

To find out more about the configuration tools available, see Section 1.13, "Oracle WebCenter Portal Administration Tools."

> **Note:** Oracle does not recommend that you edit `adf-config.xml` or `connections.xml` by hand as this can lead to misconfiguration.

While WebCenter Portal and Portal Framework applications store post deployment configuration information in MDS, configuration information for portlet producers and the discussion server is stored in the file system or the database (Table 1–5).

Table 1–5 WebCenter Portal Configuration Location

Application	Configuration Stored in MDS	Configuration Stored in File System	Configuration Stored in Database
WebCenter Portal	Yes	No	No
Portal Framework applications	Yes	No	No
Portlet producers	No	Yes	No
Discussions server	No	Yes	Yes

Discussions Server

Oracle WebCenter Portal's discussions server stores configuration information in its database. Additionally, it stores startup configuration information in *DOMAIN_HOME*`/config/fmwconfig/servers/WC_COLLABORATION/owc_discussions`. This directory contains `jive_startup.xml`, `jive.license` files, and a `logs` directory containing log files for the discussions server instance.

1.3.6 Oracle WebCenter Portal State and Configuration Persistence

WebCenter Portal and Portal Framework applications run as J2EE applications with application state and configuration persisted to the MDS repository. User session information within the application is held locally in memory. In a cluster environment, this state is replicated to other members of the cluster.

Application customizations within a portlet or service environment are persisted by that service. Out-of-the-box, Oracle portlets, any custom portlets you build, and the discussions server, all have their own database persistence mechanisms.

Analytics

WebCenter Portal's analytics capability is stateless. Requests received by analytics collectors are executed immediately. Any in-transit state, such as a request initiated by WebCenter Portal or a request processed by the analytics collector, is not guaranteed.

Activity Graph

WebCenter Portal's activity graph consists of two components:

- **Activity Graph** - does not maintain any in-memory state. The activity graph task flows query the activity graph database and display results as a list of recommendations. State is updated by the following:
  - Task flow configuration parameters
  - Personalization settings
  - "Not-interested" feature

  The first two are built on the standard Oracle Composer/Oracle ADF/MDS framework, which manages the state. The last is a feature where the user can indicate that they are not interested in a particular recommendation. This input is persisted synchronously in the database.
- **Activity Graph Engine** - runs a batch data analysis process that updates tables in the database transactionally. Although the engine does not support clustering or failover, it can recover from failure.

Administrators use the Activity Graph Scheduler to set up and monitor the nightly schedule. The results of the analysis (the recommendations) are presented through the activity graph task flows.

The Activity Graph Engine is a singleton application that has a background thread that wakes up periodically to check if it is time to run the nightly job, which can last several hours. The schedule is persisted in the database. If the managed server fails, the job continues when the managed server next starts up.

Personalization Server

WebCenter Portal's personalization server is a stateless RESTful application. All state is managed in the client requests.

1.3.7 Oracle WebCenter Portal Log File Locations

Operations performed by WebCenter Portal, Portal Framework applications, portlet producers, discussion servers, and so on, are logged directly to the WebLogic managed server where the application is running:

base_domain/servers/*Server_Name*/logs/*Server_Name*-diagnostic.log

For example, diagnostics for WebCenter Portal are logged to:
/base_domain/servers/WC_Spaces/logs/WC_Spaces-diagnostic.log

You can view the log files for each WebLogic managed server from the Oracle WebLogic Server Administration Console. To view the logs, access the Oracle WebLogic Server Administration Console http://<admin_server_host>:<port>/console, and click **Diagnostics-Log Files**.

You can also view and configure diagnostic logs through Fusion Middleware Control, see Section 28.2, "Viewing and Configuring Log Information."

1.4 WebCenter Portal

WebCenter Portal is a web-based application that offers the very latest technology for social networking, communication, collaboration, and personal productivity. Through a robust set of services and applications, WebCenter Portal brings together everything you need to exchange ideas with others, keep track of your personal and work-related tasks, interact with your critical applications, and zero in on your own projects and interests—all within a single, integrated environment.

Automatic Configuration for Tools and Services

Some tools and services are automatically configured for WebCenter Portal during the installation process. For details, see the *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*.

Default connection names are listed in Table 1–6.

Table 1–6 Connections Automatically Configured for WebCenter Portal

Tool / Service	Default Connection Name
Discussions and announcements	`WebCenterSpaces-Discussions`
Documents	`WebCenterSpaces-ucm`
Pagelet producer	`WebCenterSpaces-PageletProducer`
Personalization	`Conductor-WCPSSpaces` and `Properties-WCPSSpaces`

Preconfigured portlet producers	`wc-OmniPortlet`
	`wc-WebClipping`
	`wc-WSRPTools`

Worklists	`WebCenterSpaces-Worklist`
Portal workflows	

Configuring WebCenter Portal PostInstallation

To help you get started, see Chapter 2, "Getting WebCenter Portal Up and Running."

For information about administering WebCenter Portal, see Section 47.2, "Accessing the Portal Builder Administration Page."

1.5 Portal Framework Applications

You can develop your own portal applications using the WebCenter Portal Framework application template in JDeveloper, and deploy them to a custom WebLogic Managed Server. Portal applications built using JDeveloper are referred to as *Portal Framework applications*.

For information about developing your own portal applications, see *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

To help you get started, see:

- Chapter 4, "Getting Portal Framework Applications Up and Running"
- Chapter 5, "Maintaining Portal Framework Applications"
- Chapter 42, "Deploying Portal Framework Applications"

1.6 Planning Oracle WebCenter Portal Installations

Installing WebCenter Portal or your Portal Framework application requires a little bit of planning. Some of the questions to consider are:

- What Oracle WebCenter Portal components will be used?
- How many users will access this deployment?
- How can I provide high availability for my enterprise deployment?
- How can I secure WebCenter Portal?

For more information about planning an Oracle WebCenter Portal installation, see *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*, *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle WebCenter Portal*, and *Oracle Fusion Middleware High Availability Guide*.

1.7 Understanding the Oracle WebCenter Portal 11g Installation

The out-of-the-box topology is briefly described in Section 1.3, "Oracle WebCenter Portal Topology." Specific areas of the topology are described in the corresponding chapters, for example, security-related aspects are described in Chapter 30, "Managing Oracle WebCenter Portal Security."

For more information about Oracle WebCenter Portal installation and postinstallation administration tasks, see *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*.

> **Note:** If you want to verify which Oracle WebCenter Portal version you have installed, refer to Appendix G.2.1, "How Do I Find Out Which Oracle WebCenter Portal Version Is Installed?"

For postinstallation enterprise configuration, see *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle WebCenter Portal*.

For postinstallation high availability configuration, see *Oracle Fusion Middleware High Availability Guide*.

For postinstallation security configuration, see Chapter 30.2.5, "Post-deployment Security Configuration Tasks."

1.8 Understanding Administrative Operations, Roles, and Tools

Oracle WebCenter Portal provides several different tools with which to deploy, configure, start and stop, and maintain WebCenter Portal and Portal Framework applications. All these tools are described in Section 1.13, "Oracle WebCenter Portal Administration Tools."

Your ability to perform administration tasks depends on which Oracle WebLogic Server role you are assigned—`Admin`, `Operator`, or `Monitor`. Table 1–7 lists the Oracle WebLogic Server roles needed for common operations. These roles apply whether the operations are performed through Fusion Middleware Control, WLST commands, or the WebLogic Server Administration Console.

Table 1–7 WebCenter Portal Operations and Oracle WebLogic Server Roles

Operation	Admin Role	Operator Role	Monitor Role
WebCenter Portal and Portal Framework applications			
Start and stop	Yes	Yes	No
View performance metrics	Yes	Yes	Yes
View log information	Yes	Yes	Yes
Configure log files	Yes	Yes	Yes
View configuration	Yes	Yes	Yes
Configure new connections	Yes	Yes	No
Edit connections	Yes	Yes	No
Delete connections	Yes	Yes	No
Deploy applications	Yes	No	No
Configure security	Yes	No	No
View security (application roles/policies)	Yes	Yes	Yes
WebCenter Portal only			
Export entire application	Yes	No	No
Import entire application	Yes	No	No

Table 1–8 summarizes which tools you can use to perform various administrative operations relating to WebCenter Portal and Portal Framework applications.

Table 1–8 WebCenter Portal Operations and Administration Tools

Operation	Fusion Middleware Control	WLST Commands	WebLogic Server Admin Console	WebCenter Portal Admin / Portal Framework Application Admin
WebCenter Portal and Portal Framework applications				
Start and stop	Yes	Yes	Yes	No
View performance metrics	Yes	No	No	No
View log information	Yes	No	No	No
Configure log files	Yes	No	No	No
View configuration	Yes	Yes	No	No
Configure new connections	Yes	Yes	No	No
Edit connections	Yes	Yes	No	No
Delete connections	Yes	Yes	No	No
Manage portlet producers	Yes	Yes	No	Yes
Manage external applications	Yes	Yes	No	Yes
Deploy applications	Yes	Yes	Yes	No

Table 1–8 (Cont.) WebCenter Portal Operations and Administration Tools

Operation	Fusion Middleware Control	WLST Commands	WebLogic Server Admin Console	WebCenter Portal Admin / Portal Framework Application Admin
Configure security	Yes	Yes	Yes	No
WebCenter Portal only				
Configure workflows	Yes	Yes	No	No
Export entire application	Yes	Yes	No	No
Import entire application	Yes	Yes	No	No
Customize WebCenter Portal	No	No	No	Yes
Manage application users and roles	No	No	No	Yes
Manage pages	No	No	No	Yes
Manage portals	No	No	No	Yes
Export portals	No	No	No	Yes
Import portals	No	No	No	Yes

1.9 Performance Monitoring and Diagnostics

Performance monitoring helps administrators identify issues and performance bottlenecks in their environment. Chapter 27, "Monitoring Oracle WebCenter Portal Performance" describes the range of performance metrics available for WebCenter Portal and Portal Framework applications and how to monitor them using Fusion Middleware Control. It also describes how to troubleshoot issues by analyzing information that is recorded in diagnostic log files.

1.10 Understanding Security

The recommended security model for Oracle WebCenter Portal is based on Oracle ADF Security, which implements the Java Authentication and Authorization Service (JAAS) model. The following chapters describe security configuration for WebCenter Portal applications:

- Chapter 30, "Managing Oracle WebCenter Portal Security."
- Chapter 31, "Configuring the Identity Store"
- Chapter 32, "Configuring the Policy and Credential Store"
- Chapter 33, "Configuring Single Sign-on"
- Chapter 34, "Configuring Portal Framework Applications for Single Sign-on"
- Chapter 35, "Configuring SSL"
- Chapter 36, "Configuring WS-Security"
- Chapter 37, "Configuring Security for Portlet Producers"

1.11 WebCenter Portal Application Deployment

Chapter 42, "Deploying Portal Framework Applications" provides instructions for deploying, redeploying, and undeploying Portal Framework applications from an .EAR file created with Oracle JDeveloper.

Section 21.11, "Deploying Portlet Producer Applications" provides instructions for deploying WSRP and PDK-Java portlet producer applications.

Note: WebCenter Portal is deployed during installation. (It cannot be deployed as an .EAR file). See the "Installing Oracle WebCenter Portal" section in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*.

1.12 Data Migration, Backup, and Recovery

Oracle WebCenter Portal stores data related to its configuration and content for the various feature areas in a several locations. To facilitate disaster recovery and the full production lifecycle from development through staging and production, Oracle WebCenter Portal provides a set of utilities that enable you to back up this data, and move the data between staging and production environments.

Chapter 41, "Managing WebCenter Portal Backup, Recovery, and Cloning" describes the backup, import, and export capabilities and tools available for these tasks.

1.13 Oracle WebCenter Portal Administration Tools

Oracle offers the following tools for managing Oracle WebCenter Portal:

- Oracle Enterprise Manager Fusion Middleware Control Console
- Oracle WebLogic Server Administration Console
- Oracle WebLogic Scripting Tool (WLST)
- System MBean Browser

These administration tools apply to all applications, including WebCenter Portal, and administrators should use these tools, rather than edit configuration files, to perform administrative tasks. For help to decide which tool is best for you, see Appendix A.3, "Configuration Tools."

In addition to system administrative tools, individual applications also offer some runtime administration pages:

- WebCenter Portal - Portal Builder Administration Pages
- Portal Framework Applications - Administration Console

1.13.1 Oracle Enterprise Manager Fusion Middleware Control Console

Oracle Enterprise Manager Fusion Middleware Control Console is a browser-based management application that is deployed when you install Oracle WebCenter Portal. From Fusion Middleware Control Console, you can monitor and administer a *farm* (such as one containing Oracle WebCenter Portal, WebCenter Portal, and Portal Framework applications).

Fusion Middleware Control organizes a wide variety of performance data and administrative functions into distinct, web-based home pages. These home pages make it easy to locate the most important monitoring data and the most commonly

used administrative functions for any WebCenter Portal component—all from your web browser. For general information about the Fusion Middleware Control Console, see the "Getting Started Using Oracle Enterprise Manager Fusion Middleware Control" section in *Oracle Fusion Middleware Administrator's Guide*.

Fusion Middleware Control is the primary management tool for Oracle WebCenter Portal and can be used to:

- Deploy, undeploy, and re-deploy Portal Framework applications
- Configure back-end services and tools
- Configure security management
- Control process lifecycle
- Access log files and manage log configuration
- Manage data migration
- Monitor performance
- Diagnose run-time problems
- Manage related components, such as the parent Managed Server, MDS, portlet producers, and so on

1.13.1.1 Displaying Fusion Middleware Control Console

For information about starting Fusion Middleware Control, see Section 6.1, "Displaying Fusion Middleware Control Console."

1.13.2 Oracle WebLogic Server Administration Console

The Oracle WebLogic Server Administration Console is a browser-based, graphical user interface that you use to manage a WebLogic Server domain.

The Administration Server hosts the Administration Console, which is a Web application accessible from any supported Web browser with network access to the Administration Server Managed Servers host applications.

Use the Administration Console to:

- Configure, start, and stop WebLogic Server instances
- Configure WebLogic Server clusters
- Configure WebLogic Server services, such as database connectivity (JDBC) and messaging (JMS)
- Configure security parameters, including creating and managing users, groups, and roles
- Configure and deploy your applications
- Monitor server and application performance
- View server and domain log files
- View application deployment descriptors
- Edit selected run-time application deployment descriptor elements

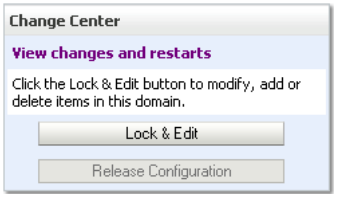
For more information about the Oracle WebLogic Server Administration Console, see the "Displaying the Oracle WebLogic Server Administration Console" section in *Oracle Fusion Middleware Administrator's Guide*.

Locking Domain Configuration

You must lock configuration settings for a domain before making any configuration changes. Navigate to the Administration Console's Change Center (Figure 1–4), and click **Lock & Edit**.

Once configuration updates are complete, release the changes by clicking **Release Configuration**.

Figure 1–4 Change Center in Oracle WebLogic Server Administration Console



1.13.3 Oracle WebLogic Scripting Tool (WLST)

Oracle provides the WebLogic Scripting Tool (WLST) to manage Oracle Fusion Middleware components, such as Oracle WebCenter Portal, from the command line.

WLST is a complete, command-line scripting environment for managing Oracle WebLogic Server domains, based on the Java scripting interpreter, Jython. In addition to supporting standard Jython features such as local variables, conditional variables, and flow control statements, WLST provides a set of scripting functions (commands) that are specific to Oracle WebLogic Server. You can extend the WebLogic scripting language to suit your needs by following the Jython language syntax.

Oracle provides WLST commands for managing connections (to content repositories, portlet producers, external applications, and other back-end services), and application migration. All Oracle WebCenter Portal WLST commands are described in the "WebCenter Portal Custom WLST Commands" chapter in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference.*

Note: If you using a third-party application server provided by a vendor other than Oracle, refer to the *Oracle Fusion Middleware Third-Party Application Server Guide* for information about third party command line tools, such as IBM WebSphere's WSADM.

1.13.3.1 Running Oracle WebLogic Scripting Tool (WLST) Commands

You *must* run all Oracle WebCenter Portal WLST commands from your **WebCenter Portal Oracle home directory** (*`WCP_ORACLE_HOME`*).

Note: If you attempt to run Oracle WebCenter Portal WLST commands from the wrong directory you will see a `NameError`. To avoid this error, always run the WLST commands from WebCenter Portal's Oracle home (*`WCP_ORACLE_HOME`*`/common/bin`) as directed below.

See also, Appendix G, "Troubleshooting Oracle WebCenter Portal."

To run WLST from the command line:

1. Navigate to your **WebCenter Portal Oracle home** directory and invoke the WLST script:

   (UNIX) *WCP_ORACLE_HOME*/common/bin/wlst.sh

   (Windows) *WCP_ORACLE_HOME*\common\bin\wlst.cmd

2. At the WLST command prompt, enter the following command to connect to the Administration Server for Oracle WebCenter Portal:

   ```
   wls:/offline>connect('user_name','password', 'host_name:port_number')
   ```

   where

   - *user_name* is the username of the operator who is connecting to the Administration Server
   - *password* is the password of the operator who is connecting to the Administration Server
   - *host_name* is the host name of the Administration Server
   - *port_number* is the port number of the Administration Server

   For example:

   ```
   connect(username='weblogic', password='mypassword',
   url='myhost.example.com:7001')
   ```

   If preferred, you can connect to the Administration Server in interactive mode without parameters:

   ```
   wls:/offline> connect()
   Please enter your username :weblogic
   Please enter your password :
   Please enter your server URL [t3://localhost:7001]:t3://myhost.example.com:7001
   Connecting to t3://myhost.example.com:7001 with userid weblogic ...
   Successfully connected to Admin Server 'AdminServer' that belongs to domain
   'wc_domain'.
   ```

   For help with this command, type `help('connect')` at the WLST command prompt.

   Note: If SSL is enabled, you must edit the `wlst.sh` or `wlst.cmd` file and append the following to `JVM_ARGS`:

   ```
   -Dweblogic.security.SSL.ignoreHostnameVerification=true
   -Dweblogic.security.TrustKeyStore=DemoTrust
   ```

   or `setenv CONFIG_JVM_ARGS`

   ```
   -Dweblogic.security.SSL.ignoreHostnameVerification=true
   -Dweblogic.security.TrustKeyStore=DemoTrust
   ```

   ---

3. Once connected to the Administration Server you can run Oracle WebCenter Portal WLST commands, and any other generic WLST command.

Hints and Tips Running for Oracle WebCenter Portal WLST Commands

- **To list Oracle WebCenter Portal WLST commands**, type: `help('webcenter')` at the WLST command prompt.

If the message `No help for webcenter found...` displays, you are probably running the WLST script from the wrong directory, for example, you might be running `wlst.sh` or `wlst.cmd` from the `oracle_common` directory instead of *`WCP_ORACLE_HOME`*`/common/bin`.

- **For help on a particular command**, type: `help('`*`WLST_command_name`*`')` at the WLST command prompt.
- **Include argument names when running commands** and especially when writing WLST scripts. For example, it is good practice to enter:

  ```
  createExtAppConnection(appName='webcenter', name='myXApp'...
  ```

  rather than:

  ```
  createExtAppConnection('webcenter', 'myXApp'...
  ```

  Either syntax is valid but when you include the argument names, errors and misconfiguration is less likely. Also, if arguments are added in the future, the command does not fail or configure the wrong property.
- **In a clustered environment, remember to specify the "server" argument when running commands**. All Oracle WebCenter Portal WLST commands include a `server` argument which becomes mandatory when WebCenter Portal or Portal Framework applications are deployed to cluster. See also, the "WebCenter Portal Custom WLST Commands" chapter in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference.*
- **Online documentation for Oracle WebCenter Portal WLST commands** is available from the "WebCenter Portal Custom WLST Commands" chapter in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference.*

1.13.4 System MBean Browser

Fusion Middleware Control provides a set of MBean browsers that allow to you browse the MBeans for an Oracle WebLogic Server or for a selected application.

> **Note:** While you can monitor and configure WebCenter Portal and Portal Framework application MBeans from the System MBean browser, it is not the preferred tool for configuration. Oracle recommends that you configure applications using WLST commands or through the WebCenter Portal Settings menu options in Fusion Middleware Control (available from the application's home page).

To access application MBeans:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal or your Portal Framework application. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **System MBean Browser**.

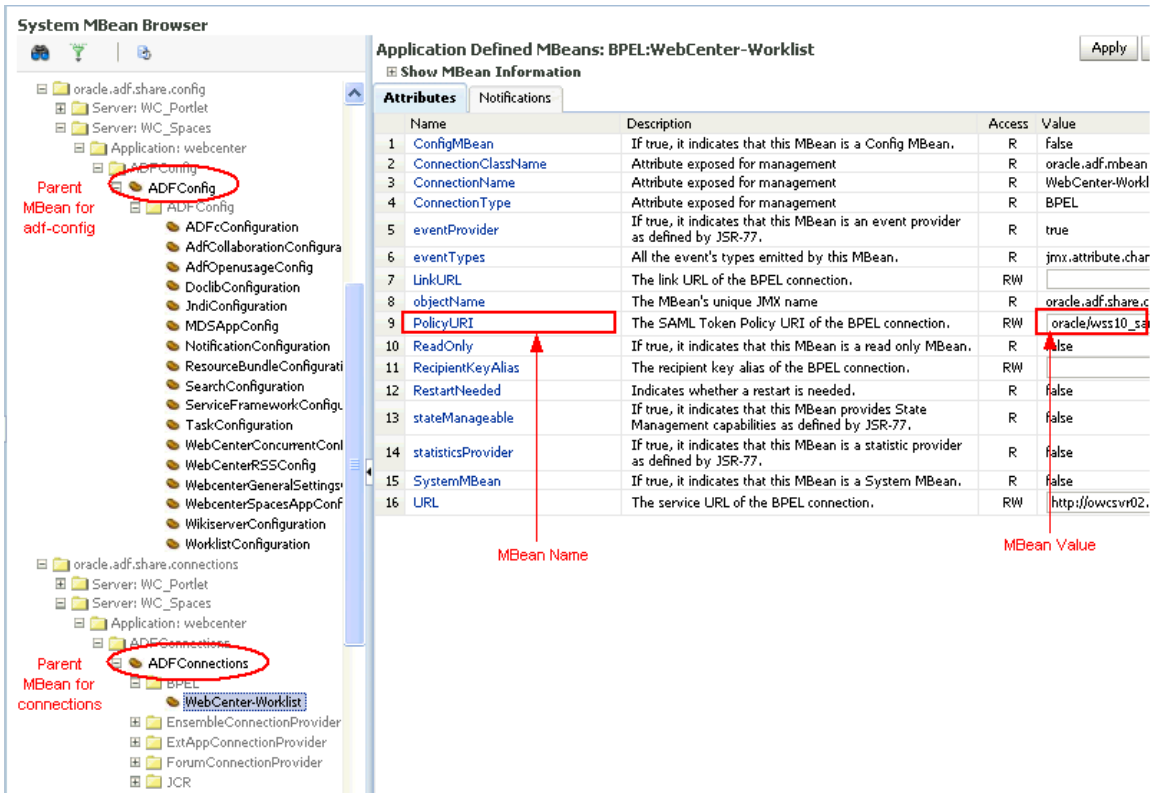
- For Portal Framework applications - From the **Application Deployment** menu, select **System MBean Browser**.

3. Expand **Application Defined MBeans**.
4. Navigate to the MBean you want to view or configure.

   For example, for a Portal Framework applications, you might want to navigate to MBeans for `adf-config.xml` and `connections.xml` as follows (Figure 1–5):

   - `adf-config` - Click **oracle.adf.share.config** >**Server: name** >**Application: name** >**ADFConfig** >**ADFConfig** >**ADFConfig**
   - `connections` - Click **oracle.adf.share.connections** >**Server: name** >**Application: name** >**ADFConnnections** >**ADFConnections**
5. To view an MBean's attributes, select the **Attributes** tab. Some attributes allow you to change their values. To do so, enter the value in the **Value** column.

Figure 1–5 Systems MBean Browser



6. Click **Apply** to update attribute values.
7. Navigate to the parent MBean (for example, **ADFConfig** or **ADFConnections**), select the **Operations** tab, and click **save** to save the changes.
8. Restart the managed server on which WebCenter Portal or the Portal Framework application is deployed. For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

1.13.5 WebCenter Portal - Portal Builder Administration Pages

WebCenter Portal provides several administration pages of its own. The Portal Builder administration pages appear only to users who have logged in to the application using an administrator user name and password.

Portal Builder administration pages allow you to:

- Customize WebCenter Portal
- Manage users and roles
- Manage tool and service settings
- Manage portlet producers and external applications
- Manage individual portals and portal templates
- Create and manage business role pages
- Manage personal pages
- Export and import individual portals and portal templates

For more details, see Section 47.2, "Accessing the Portal Builder Administration Page."

1.13.6 Portal Framework Applications - Administration Console

Portal applications built using the WebCenter Portal Framework application template in JDeveloper can also include administration pages that enable administrators to perform common administrative duties at runtime. For more information, see Chapter 1, "Introduction to Administering Oracle WebCenter Portal."

Part II

Getting Started With WebCenter Portal Administration

This part of *Oracle Fusion Middleware Administering Oracle WebCenter Portal* provides checklists to help you get started with WebCenter Portal administration.

Part II contains the following chapters:

- Chapter 2, "Getting WebCenter Portal Up and Running"
- Chapter 3, "Maintaining WebCenter Portal"

2

Getting WebCenter Portal Up and Running

This chapter describes the roles and responsibilities of system administrators to get WebCenter Portal up and running.

This chapter includes the following topics:

- Section 2.1, "Role of the System Administrator"
- Section 2.2, "Installing WebCenter Portal"
- Section 2.3, "Setting Up WebCenter Portal for the First Time (Roadmap)"
- Section 2.4, "Customizing WebCenter Portal for the First Time (Roadmap)"

System administrators working with Portal Framework applications, should refer to Chapter 4, "Getting Portal Framework Applications Up and Running."

Permissions: To perform the tasks in this chapter, you must be granted the following roles:

- **WebLogic Server:** `Admin` role granted through the Oracle WebLogic Server Administration Console.

  Users with this role are also known as *Fusion Middleware administrators*.

- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.

  Users with this role are also known as *WebCenter Portal administrators*.

See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

2.1 Role of the System Administrator

Oracle Fusion Middleware provides a single administrative role with *complete* administrative capabilities—the `Admin` role. System administrators with this role can perform the complete range of security-sensitive administrative duties, and all installation, configuration, and audit tasks. This administrator is also responsible for setting up and configuring WebCenter Portal immediately after installation, and performing on-going administrative tasks for WebCenter Portal and other Oracle WebCenter Portal components. This administrator is sometimes known as the *Fusion Middleware administrator*.

During installation, a single system administrator account is created named `weblogic`. The password is the one provided during installation.

Use this administrator account to log in to the Fusion Middleware Control Console and WebCenter Portal, and assign administrative privileges to other users:

- **Fusion Middleware Control** - Add one more users to the `Administrator` group using the Oracle WebLogic Administration Console or Oracle WebLogic Scripting Tool (WLST). For details, see the "Administrative Users and Roles" section in *Oracle Fusion Middleware Application Security Guide*.

  Oracle WebLogic Server provides two other roles, in addition to the `Admin` role, namely `Operator` and `Monitor`. To find out more about these role, see Table 1–7, " WebCenter Portal Operations and Oracle WebLogic Server Roles" in Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

- **WebCenter Portal** - Assign one more users the `Administrator` role through Portal Builder Administration. For details, see Section 49.5.3, "Giving a User Administrative Privileges."

  WebCenter Portal administrators have the highest privileges within the WebCenter Portal application. This administrator can view and customize every aspect of the WebCenter Portal, manage users and roles, and delegate responsibilities to others.

To find out what other tasks system administrators must do to get WebCenter Portal up and running, follow the steps listed under Table 2–1, " Roadmap - Setting Up WebCenter Portal for the First Time".

To find out what system administrator can do to customize WebCenter Portal out-of-the-box, follow the Table 2–2, " Roadmap - Customizing WebCenter Portal for the First Time".

Note: System administrators are also responsible for all on-going administrative tasks, for details see Section 3.2, "System Administration for WebCenter Portal (Roadmap)."

2.2 Installing WebCenter Portal

WebCenter Portal installation is described in the *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*.

2.3 Setting Up WebCenter Portal for the First Time (Roadmap)

The flow chart depicted in Figure 2–1 and Table 2–1 in this section provide an overview of the tasks required to get WebCenter Portal up and running.

Figure 2–1 Setting Up WebCenter Portal for the First Time

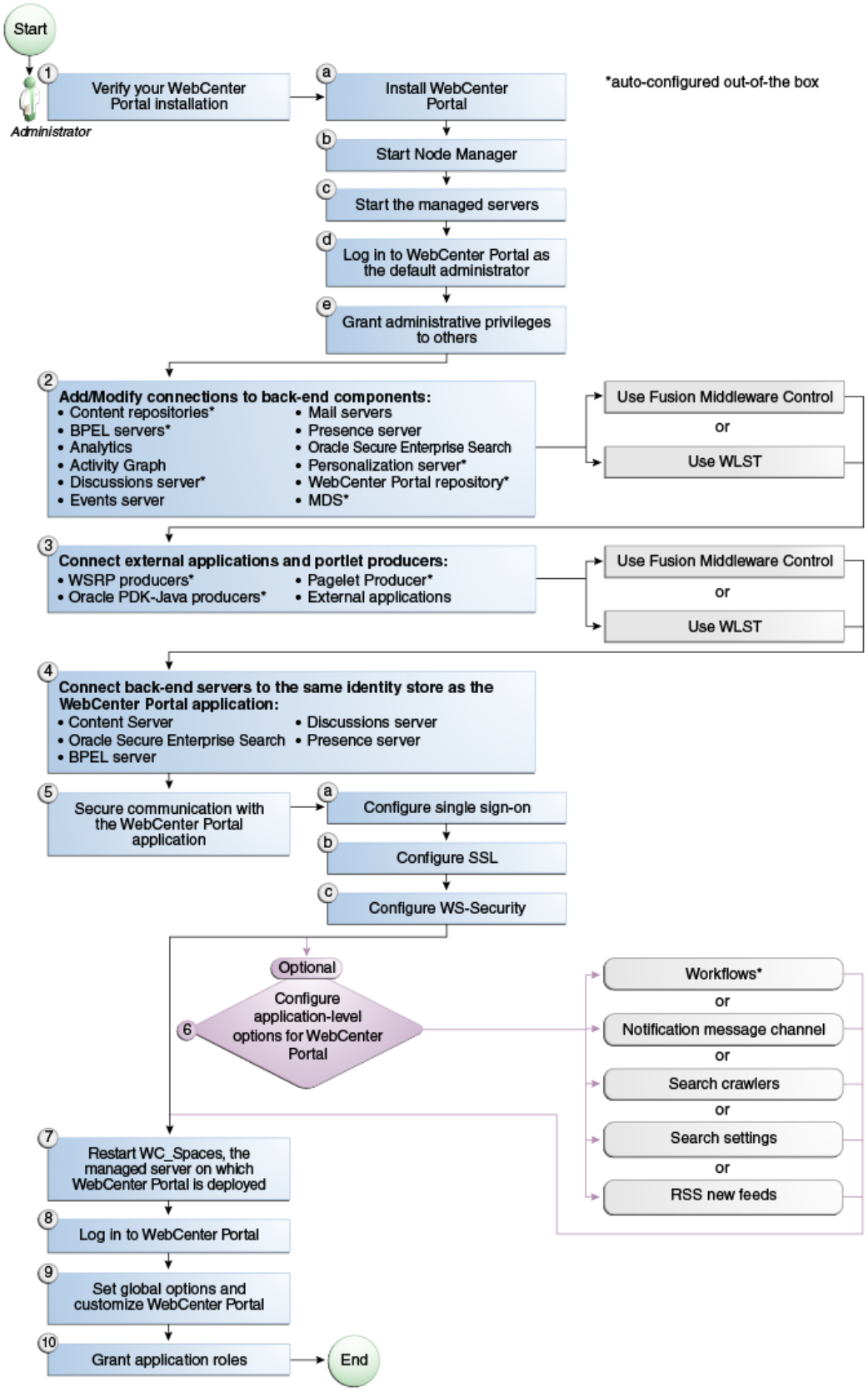


Table 2–1 Roadmap - Setting Up WebCenter Portal for the First Time

Actor	Task	Sub-task	Notes
Fusion Middleware Administrator	**1.** Verify your WebCenter Portal installation	**1.a** Install WebCenter Portal **1.b** Start Node Manager **1.c** Start the managed servers **1.d** Log in to WebCenter Portal as the default administrator **1.e** Giving a User Administrative Privileges	
Fusion Middleware Administrator	**2.** Add/modify connections to back-end components using either of the following tools: ■ Fusion Middleware Control ■ WLST		Back-end components may include: ■ Content repositories[1] ■ BPEL servers[1] ■ Analytics collector ■ Activity graph engines ■ Discussions server[1] ■ Events server ■ Mail servers ■ Presence server ■ Oracle Secure Enterprise Search ■ Personalization server[1] ■ WebCenter Portal repository[1] ■ MDS[1]
Fusion Middleware Administrator	**3.** Connect external applications and portlet producers using either of the following tools: ■ Fusion Middleware Control ■ WLST		Portlet producers may include: ■ WSRP producers[1] ■ Oracle PDK-Java producers[1] ■ Pagelet producer[1]

Table 2–1 (Cont.) Roadmap - Setting Up WebCenter Portal for the First Time

Actor	Task	Sub-task	Notes
Fusion Middleware Administrator	**4.** Connect back-end severs to the same identity store WebCenter Portal		Back-end servers may include: ■ Oracle WebCenter Content Server ■ Oracle Secure Enterprise Search ■ BPEL server ■ Discussions server ■ Presence server
Fusion Middleware Administrator	**5.** Secure communication with WebCenter Portal	**5.a** Configure single sign-on **5.b** Configure SSL **5.c** Configure WS-Security	
Fusion Middleware Administrator	**6.** (Optional) Configure system options for WebCenter Portal: ■ Portal workflows[1] ■ Notification message channel ■ Search crawlers ■ Search settings ■ RSS news feeds ■ Microsoft Office integration		
Fusion Middleware Administrator	**7.** Restart WC_Spaces, the managed server on which WebCenter Portal is deployed		
WebCenter Portal Administrator	**8.** Log in to WebCenter Portal		
WebCenter Portal Administrator	**9.** Set global options and customize WebCenter Portal		
WebCenter Portal Administrator	**10.** Assigning Users (and Groups) to Roles		

[1] Auto-configured out-of-the-box

2.4 Customizing WebCenter Portal for the First Time (Roadmap)

The roadmap in Table 2–2 outlines the tasks that a WebCenter Portal administrator might perform to customize WebCenter Portal for a new target audience.

Table 2–2 Roadmap - Customizing WebCenter Portal for the First Time

Task	Documentation	Actor
1. Log in to WebCenter Portal	Log in to WebCenter Portal with administrative privileges and access the administration pages: ■ Accessing the Portal Builder Administration Page **Tips:** WebCenter Portal URL is `http://<host>:<port>/webcenter` Portal Builder Administration URL is `http://host:port/webcenter/portal/builder/administration`	WebCenter Portal Admin
2. Customize WebCenter Portal	Customize WebCenter Portal to suit your audience. Choose a name and logo for your application, apply a corporate brand, set language options, choose default portals, default assets, and more. For details, see: ■ Performing Portal Builder Administration Tasks ■ Configuring Global Defaults Across Portals ■ Customizing System Pages ■ Managing Business Role Pages ■ Managing Personal Pages	WebCenter Portal Admin
3. Determine self-registration policy	Establish your policy regarding new user registration. Allow users outside of the WebCenter Portal community to self -register on an invitation-only basis or extend self-registration to the public: ■ Enabling Self-Registration By Invitation-Only ■ Enabling Anyone to Self-Register	WebCenter Portal Admin
4. Plan the public user experience	First impressions are extremely important. Determine the content displayed on your Welcome page and the appearance of WebCenter Portal before users login: ■ Customizing the Welcome Page ■ Customizing the Login Page ■ Customizing the Self-Registration Page ■ Choosing a Default Display Language ■ Granting Permissions to the Public-User	WebCenter Portal Admin
5. Create roles and delegate responsibilities to other users	Create roles to characterize groups of users and determine what they can see and do in WebCenter Portal. Manage and assign roles for any user in the identity store: ■ About Portal Security ■ Assigning Users (and Groups) to Roles ■ Defining Application Roles ■ Giving a User Administrative Privileges ■ Modifying Application Role Permissions	WebCenter Portal Admin

Table 2–2 (Cont.) Roadmap - Customizing WebCenter Portal for the First Time

Task	Documentation	Actor
6. Customize the Home portal	Design the default Home portal for WebCenter Portal users. Give them instant access to important information and applications relevant to their roles: ■ Setting Page Creation Defaults for Business Role Pages ■ Creating a Business Role Page Encourage or enforce a consistent look and feel through default page schemes and default page templates: ■ Choosing a Default Look and Feel for New Pages	WebCenter Portal Admin
7. Set up discussion forums and announcements	Configure default options for discussion forums and announcements: ■ Configuring Discussion Forum Options for WebCenter Portal	WebCenter Portal Admin
8. Set up people connection components	Configure defaults for activity streams, personal profiles, connections, messages boards, and feedback: ■ Configuring People Connections for WebCenter Portal	WebCenter Portal Admin
9. Set up mail notifications	Configure default options for everyone's mail: ■ Configuring Send Mail Notifications	WebCenter Portal Admin
10. Provide ready-made portals and portal templates	Users can create and manage their own portals without centralized administration. Give them a head-start by creating templates for the types of workspaces and communities they are likely to build: ■ Creating and Building a New Portal ■ Creating a New Portal Template	WebCenter Portal Admin

3

Maintaining WebCenter Portal

This chapter describes the roles and responsibilities of system administrators to keep WebCenter Portal up and running.

This chapter includes the following topics:

- Section 3.1, "Role of the System Administrator"
- Section 3.2, "System Administration for WebCenter Portal (Roadmap)"
- Section 3.3, "System Administration for Portal Builder (Roadmap)"

System administrators maintaining Portal Framework applications should refer to Chapter 5, "Maintaining Portal Framework Applications."

Permissions: To perform the tasks in this chapter, you must be granted the following roles:

- **WebLogic Server:** `Admin` role granted through the Oracle WebLogic Server Administration Console.

  Users with this role are also known as *Fusion Middleware administrators*.

- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.

  Users with this role are also known as *WebCenter Portal administrators*.

See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

3.1 Role of the System Administrator

Oracle Fusion Middleware provides a single administrative role with complete administrative capabilities—the `Admin` role. System administrator with this role can perform the complete range of security-sensitive administrative duties, and all installation, configuration, and audit tasks. This administrator is also responsible for setting up and configuring WebCenter Portal immediately after installation, and performing on-going administrative tasks for WebCenter Portal and other Oracle WebCenter Portal components. This administrator is sometimes known as the *Fusion Middleware administrator*.

A single system administrator account (`weblogic` by default) is set up when Fusion Middleware is installed. The password is the one you provided during installation.

This default system administrator is the only user assigned to the WebCenter Portal `Administrator` role. WebCenter Portal administrators have the highest application privileges within the WebCenter Portal application itself. This administrator can view and customize every aspect of WebCenter Portal, manage users and roles, and delegate responsibilities to others.

To find out what on-going administrative tasks a system administrator is expected to perform for WebCenter Portal, follow the roadmaps in Table 3–1, " Roadmap - Administering and Monitoring WebCenter Portal" and Roadmap - Keeping WebCenter Portal Up and Running.

Note: The system administrator is also responsible for getting WebCenter Portal up and running out-of-the-box and customizing WebCenter Portal out-of-the-box. For details see:

- Section 2.3, "Setting Up WebCenter Portal for the First Time (Roadmap)."
- Section 2.4, "Customizing WebCenter Portal for the First Time (Roadmap)"

3.2 System Administration for WebCenter Portal (Roadmap)

The roadmap in Table 3–1 outlines typical tasks that a system administrator might perform to keep WebCenter Portal up and running.

Table 3–1 Roadmap - Administering and Monitoring WebCenter Portal

Task	Documentation	Role
Stop and start the managed server	Restart the managed server on which the WebCenter Portal application is deployed to effect configuration changes or for routine maintenance: ▪ Starting and Stopping Managed Servers for WebCenter Portal Application Deployments Tip: The managed server for WebCenter Portal is named `WC_Spaces`.	Fusion Middleware Admin
View and manage log files	Identify and diagnose problems through log files. WebCenter Portal logs record all types of events, including startup and shutdown information, errors, warnings, and other information: ▪ Viewing and Configuring WebCenter Portal Logs	Fusion Middleware Admin

Table 3–1 (Cont.) Roadmap - Administering and Monitoring WebCenter Portal

Task	Documentation	Role
Monitor performance	Analyze the performance of the WebCenter Portal application and monitor its current status through Fusion Middleware Control Console: ▪ Viewing Performance Metrics Using Fusion Middleware Control ▪ Monitoring WebCenter Portal ▪ Using Key Performance Metric Data to Analyze and Diagnose System Health	Fusion Middleware Admin
	System administrators granted one of these WebLogic Server roles can view performance metrics: `Admin`, `Operator`, `Monitor`. To find out more, see Section 1.8, "Understanding Administrative Operations, Roles, and Tools.".	WebCenter Portal Admin
	WebCenter Portal administrators can monitor application performance and usage using WebCenter Portal's analytics feature: ▪ Understanding the Analytics Administration Page in WebCenter Portal	
Tune application properties	Reconfigure performance related parameters for the WebCenter Portal environment, WebCenter Portal application, and WebCenter Portal components: ▪ Tuning Oracle WebCenter Portal Performance	Fusion Middleware Admin
Stop and start WebCenter Portal	System administrators may shut down WebCenter Portal for maintenance purposes and then restart the application: ▪ Starting WebCenter Portal Using Fusion Middleware Control ▪ Stopping WebCenter Portal Using Fusion Middleware Control	Fusion Middleware Admin
Modify back-end services	Add, modify, and delete connections through Fusion Middleware Control Console. See:	Fusion Middleware Admin
▪ **Content repositories**	▪ Managing Content Repositories	
▪ **Mail servers**	▪ Managing Mail	
▪ **BPEL servers**	▪ Managing Worklists	
▪ **Collaboration**	▪ Managing Announcements and Discussions ▪ Managing Instant Messaging and Presence	
▪ **Calendar**	▪ Managing Calendar Events	
▪ **Secure Enterprise Search**	▪ Managing Oracle Secure Enterprise Search in WebCenter Portal	
▪ **Analytics**	▪ Managing Analytics	
▪ **Activity Graph**	▪ Managing Activity Graph	
▪ **Personalization**	▪ Managing Personalization	
▪ **Events, Links, Lists, Notes, Tags, and People Connections**	▪ Setting Up Database Connections ▪ Setting Up the MDS Repository	

Table 3–1 (Cont.) Roadmap - Administering and Monitoring WebCenter Portal

Task	Documentation	Role
Modify external applications and portlet producers ■ **External Applications** ■ **Portlet Producers**	Add, modify, and delete connections through Fusion Middleware Control Console. See: ■ Managing External Applications ■ Registering WSRP Producers ■ Registering Oracle PDK-Java Producers ■ Registering Pagelet Producer	Fusion Middleware Admin
Configure SSL communication	Configure secure communication: ■ Configuring SSL ■ Configuring WS-Security ■ Configuring Single Sign-on See also *Oracle Fusion Middleware Application Security Guide*.	Fusion Middleware Admin
Reassociate your identity, policy, and credential stores	Reassociate your identity or policy stores: ■ Configuring the Identity Store ■ Configuring the Policy and Credential Store See also *Oracle Fusion Middleware Application Security Guide*.	Fusion Middleware Admin
Reconfigure WebCenter Portal repository	Reconfigure the WebCenter Portal repository: ■ Setting Up Database Connections	Fusion Middleware Admin
Reconfigure MDS repository	Reconfigure the application's MDS repository: ■ Setting Up the MDS Repository See also *Oracle Fusion Middleware Administrator's Guide*: ■ Managing the MDS Repository ■ Configuring an Application to Use a Different MDS Repository or Partition ■ Moving Metadata from a Source System to a Target System	Fusion Middleware Admin
Reconfigure WebCenter Portal workflows	Install WebCenter Portal workflows on a different BPEL server and reconfigure the connection: ■ Installing WebCenter Portal Workflows ■ Specifying the BPEL Server Hosting WebCenter Portal Workflows	Fusion Middleware Admin
Migrate or export portals, portal templates, assets, or the entire portal server	Use various export facilities to move content to a remote instance or between stage and production environments: ■ Exporting WebCenter Portal to an Archive ■ Deploying Portals ■ Deploying Portal Templates ■ Deploying Assets ■ Deploying Devices and Device Groups	Fusion Middleware Admin

Table 3–1 (Cont.) Roadmap - Administering and Monitoring WebCenter Portal

Task	Documentation	Role
Import WebCenter Portal application	Use various import facilities to restore WebCenter Portal from a backup or to move content to a remote instance or between stage and production environments: ▪ Importing a WebCenter Portal Archive ▪ Deploying Portals ▪ Deploying Portal Templates ▪ Deploying Assets ▪ Deploying Devices and Device Groups	Fusion Middleware Admin

3.3 System Administration for Portal Builder (Roadmap)

The roadmap in Table 3–2 outlines typical tasks that a system administrator might perform while Portal Builder is up and running.

If Portal Builder must be taken offline for maintenance, ensure that a suitable message displays to any users who attempt to access the application while it is offline.

Table 3–2 Roadmap - Keeping WebCenter Portal Up and Running

Task	Documentation	Role
Modify application Settings	Modify application-wide settings as required: ▪ Performing Portal Builder Administration Tasks ▪ Configuring Global Defaults Across Portals ▪ Managing Tools and Services ▪ Customizing System Pages ▪ Managing Business Role Pages ▪ Managing Personal Pages	WebCenter Portal Admin
Manage Home portal	Manage personal pages and business role pages. Push content to the Home portal: ▪ Managing Business Role Pages ▪ Managing Personal Pages ▪ Customizing System Pages	WebCenter Portal Admin
Manage portals	Take any portal temporarily offline and close down any portal that is inactive. Edit and delete any portal: ▪ Viewing Information About Any Portal ▪ Closing Any Portal ▪ Taking Any Portal Offline ▪ Bringing Any Portal Back Online ▪ Deleting a Portal	WebCenter Portal Admin
Manage portal templates	Manage portal templates. Review and delete any template: ▪ Creating a New Portal Template	WebCenter Portal Admin
Maintain users and roles	Maintain security. Modify user role permissions and assign new roles: ▪ Modifying Application Role Permissions ▪ Assigning a User to a Different Role	WebCenter Portal Admin

Table 3–2 (Cont.) Roadmap - Keeping WebCenter Portal Up and Running

Task	Documentation	Role
Manage external applications	Maintain external applications. Add, modify, and delete entries: ■ Registering External Applications	WebCenter Portal Admin AppConnectionManager
Manage portlet producers	Maintain portlet producers. Add, modify, and delete entries: ■ Registering Portlet Producers	WebCenter Portal Admin AppConnectionManager

Part III

Getting Started With Portal Framework Application Administration

This part of *Oracle Fusion Middleware Administering Oracle WebCenter Portal* provides checklists to help you get started with Portal Framework application administration.

Part III contains the following chapters:

- Chapter 4, "Getting Portal Framework Applications Up and Running"
- Chapter 5, "Maintaining Portal Framework Applications"

4

Getting Portal Framework Applications Up and Running

The chapter describes what system administrators must do, after installation, to get Portal Framework applications up and running. Portal Framework applications are portal applications built using the WebCenter Portal Framework application template in Oracle JDeveloper.

The chapter includes the following topics:

- Section 4.1, "Installing Oracle WebCenter Portal and the WebCenter Portal Framework Libraries"
- Section 4.2, "Deploying Portal Framework Applications for the First Time (Roadmap)"

Permissions: To perform the tasks in this chapter, you must be granted the following roles:

- **WebLogic Server:** `Admin` role granted through the Oracle WebLogic Server Administration Console.

  Users with this role are also known as *Fusion Middleware administrators*.

- **Portal Framework application**: `Administrator` role granted through the Administration Console.

  Users with this role are also known as *Portal Framework administrators*.

See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

Oracle WebCenter Portal's out-of-the-box portal application "*WebCenter Portal*" requires some special administration tasks that your own Portal Framework applications do not. To see a comprehensive list of these tasks, refer to Chapter 2, "Getting WebCenter Portal Up and Running."

4.1 Installing Oracle WebCenter Portal and the WebCenter Portal Framework Libraries

Oracle WebCenter Portal installation is described in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*.

Oracle JDeveloper installation, required for building Portal Framework applications, is described in *Oracle Fusion Middleware Installation Guide for Oracle JDeveloper*.

Portal Framework applications can be deployed to any WebLogic Server instance that is provisioned with WebCenter Portal's Framework shared library files. For details, see, Section 42.1.4, "Creating a Managed Server."

4.2 Deploying Portal Framework Applications for the First Time (Roadmap)

The roadmap in Table 4–1 outlines the tasks that a system administrator must perform to deploy a Portal Framework application, and get it up and running.

> **Note:** Oracle WebCenter Portal's out-of-the-box portal application "*WebCenter Portal*" requires additional administration tasks that other Portal Framework applications do not. To see a comprehensive list of these tasks, refer to Chapter 2, "Getting WebCenter Portal Up and Running."

Table 4–1 Roadmap - Getting Portal Framework Applications Up and Running for the First Time

Step	Documentation	Role
Step 1 - Verify your Oracle WebCenter Portal installation	Verify your Oracle WebCenter Portal installation and settings. See: ▪ Installing Oracle WebCenter Portal and the WebCenter Portal Framework Libraries ▪ Starting Node Manager	Fusion Middleware Admin
Step 2 - Launch Fusion Middleware Control	Launch the Fusion Middleware Control Console, a Web-based management tool for WebCenter Portal applications. See: ▪ Displaying Fusion Middleware Control Console ▪ Navigating to the Home Page for Portal Framework Applications Learn about the command-line administration tool WLST. See Section 1.13.3, "Oracle WebLogic Scripting Tool (WLST)."	Fusion Middleware Admin
Step 3 - Deploy the Portal Framework application	Create a suitable container in which to deploy the Portal framework application archive: ▪ Creating a Managed Server ▪ Creating and Registering the Metadata Service Repository ▪ Deploying the Application to a WebLogic Managed Server See also, Section 42, "Deploying Portal Framework Applications."	Fusion Middleware Admin
Step 4 - Reconfigure back-end servers	Reconfigure back-end server connections, if required, through Fusion Middleware Control.	Fusion Middleware Admin
▪ **Content Repositories**	▪ Managing Content Repositories	

Table 4–1 (Cont.) Roadmap - Getting Portal Framework Applications Up and Running for the First Time

Step	Documentation	Role
■ **Mail Servers** ■ **BPEL Servers** ■ **Collaboration** ■ **Secure Enterprise Search** ■ **Analytics** ■ **Activity Graph** ■ **External Applications** ■ **Portlet Producers** ■ **People Connections, Group Events, Links, Lists, Notes, and Tags**	■ Managing Mail ■ Managing Worklists ■ Managing Announcements and Discussions ■ Managing Instant Messaging and Presence ■ Managing Subscriptions and Notifications ■ Managing Oracle Secure Enterprise Search in WebCenter Portal ■ Managing Analytics ■ Managing Activity Graph ■ Managing External Applications ■ Registering WSRP Producers ■ Registering Oracle PDK-Java Producers ■ Registering Pagelet Producer ■ Setting Up Database Connections ■ Setting Up the MDS Repository	
Step 5 - Connect to an identity store	Ensure that your identity store is installed, configured, and contains all the required user data. See: ■ Configuring the Identity Store See also *Oracle Fusion Middleware Application Security Guide*.	Fusion Middleware Admin
Step 6 - Restart the managed server	Restart the managed server on which the application is deployed. See: ■ Starting and Stopping Managed Servers for WebCenter Portal Application Deployments	Fusion Middleware Admin
Step 7 - Verify Portal Framework application configuration	Login to the application to verify the configuration: identity store, tools/services, applications, and so on. ■ Administering Portal Framework Applications Using the Administration Console	Portal Framework Application Admin
Step 8 - Perform administrative tasks through the application's Administration Console	Perform administrative duties: ■ Set application-level preferences ■ Manage users and grant application roles ■ Manage and configure application assets ■ Manage and configure content ■ Manage and configure portlet producers ■ Manage and configure external applications ■ Create and manage polls See: Administering Portal Framework Applications Using the Administration Console	Portal Framework Application Admin

5 Maintaining Portal Framework Applications

The chapter describes what system administrators can do to keep Portal Framework applications up and running.

This chapter includes the following topic:

- Section 5.1, "System Administration for Portal Framework Applications (Roadmap)"

Permissions: To perform the tasks in this chapter, you must be granted the following roles:

- **WebLogic Server:** `Admin` role granted through the Oracle WebLogic Server Administration Console.

  Users with this role are also known as *Fusion Middleware administrators*.

- **Portal Framework application**: `Administrator` role granted through the Administration Console.

  Users with this role are also known as *Portal Framework administrators*.

See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

Oracle WebCenter Portal's out-of-the-box portal application "*WebCenter Portal*" requires some special maintenance tasks that your own Portal Framework applications do not. To see a comprehensive list of these tasks, see Chapter 3, "Maintaining WebCenter Portal."

5.1 System Administration for Portal Framework Applications (Roadmap)

The roadmap in Table 5–1 outlines typical tasks that a system administrator might perform to keep a Portal Framework application up and running.

If the Portal Framework application must temporarily shut down for maintenance, ensure that a suitable message displays to any users who attempt to access the application while it is offline.

Table 5–1 Roadmap - Maintaining Portal Framework Applications

Tasks	Documentation	Role
Stop and start the managed server	Restart the managed server on which the Portal Framework application is deployed to effect configuration changes or for routine maintenance: ▪ Starting and Stopping Managed Servers for WebCenter Portal Application Deployments	Fusion Middleware Admin
Stop and start the Portal Framework application	Shut down the application for maintenance purposes and then restart the application: ▪ Starting and Stopping Portal Framework Applications	Fusion Middleware Admin
Maintain back-end services	Add, modify, and delete connections through the Fusion Middleware Control Console:	Fusion Middleware Admin
▪ **Content Repositories**	▪ Managing Content Repositories	
▪ **Mail Servers**	▪ Managing Mail	
▪ **BPEL Servers**	▪ Managing Worklists	
▪ **Collaboration**	▪ Managing Announcements and Discussions ▪ Managing Instant Messaging and Presence	
▪ **Secure Enterprise Search**	▪ Managing Oracle Secure Enterprise Search in WebCenter Portal	
▪ **Analytics**	▪ Managing Analytics	
▪ **Activity Graph**	▪ Managing Activity Graph	
Maintain external applications and portlet producers	Add, modify, and delete connections through Oracle Enterprise Manager Fusion Middleware Control Console. See:	Fusion Middleware Admin
▪ **External Applications**	▪ Managing External Applications	
▪ **Portlet Producers**	▪ Registering WSRP Producers ▪ Registering Oracle PDK-Java Producers ▪ Registering Pagelet Producer	
Reassociate your identity, policy and credential stores	Reassociate your identity or policy stores: ▪ Configuring the Identity Store ▪ Configuring the Policy and Credential Store See also *Oracle Fusion Middleware Application Security Guide*.	Fusion Middleware Admin
Reconfigure the MDS repository	▪ Setting Up the MDS Repository	Fusion Middleware Admin
Reconfigure WebCenter Portal repository	▪ Setting Up Database Connections	

Table 5–1 (Cont.) Roadmap - Maintaining Portal Framework Applications

Tasks	Documentation	Role
Export Portal Framework application data	Migrate data to a remote instance or between stage and production environments: ■ Exporting Metadata for Portal Framework Applications ■ Exporting Portlet Client Metadata for Portal Framework Applications ■ Migrating Security for Portal Framework Applications ■ Migrating Schema Data for Portal Framework Applications See also, "Managing Export, Import, Backup, and Recovery for Portal Framework Applications"	Fusion Middleware Admin
Import Portal Framework application data	Use the import facility to move content to a remote instance or between stage and production environments: ■ Importing Metadata for Portal Framework Applications ■ Importing Portlet Client Metadata for Portal Framework Applications ■ Migrating Security for Portal Framework Applications ■ Migrating Schema Data for Portal Framework Applications	Fusion Middleware Admin
View and manage log files	Identify and diagnose problems through log files. Portal Framework application logs record all types of events, including startup and shutdown information, errors, warnings, and other information: ■ Viewing and Configuring Portal Framework Application Logs	Fusion Middleware Admin
Monitor performance	Analyze the performance of the Portal Framework application and monitor its current status through Fusion Middleware Control Console: ■ Viewing Performance Metrics Using Fusion Middleware Control ■ Monitoring a Portal Framework Application	Fusion Middleware Admin
Tune application properties	Reconfigure performance related parameters for the WebCenter Portal environment, application, and tools/services: ■ Tuning Oracle WebCenter Portal Performance	Fusion Middleware Admin
Perform application administrative tasks through the application's Administration Console	Perform application administrative duties: ■ Set application-level preferences ■ Manage users and grant application roles ■ Manage and configure application assets ■ Manage and configure content ■ Manage and configure portlet producers ■ Manage and configure external applications ■ Create and manage polls See: Administering Portal Framework Applications Using the Administration Console	Portal Framework Application Admin

Part IV

Basic System Administration

This part of Oracle Fusion Middleware Administering Oracle WebCenter Portal presents basic system administration tasks for WebCenter Portal and WebCenter Portal Framework applications.

Part IV contains the following chapters:

- Chapter 6, "Starting Enterprise Manager Fusion Middleware Control"
- Chapter 7, "Starting and Stopping Managed Servers and Applications for Oracle WebCenter Portal"

6

Starting Enterprise Manager Fusion Middleware Control

This chapter describes how to access Oracle Enterprise Manager Fusion Middleware Control Console, and display Oracle WebCenter Portal-related pages from where you can perform all necessary configuration, monitoring, and management tasks.

This chapter includes the following topics:

- Section 6.1, "Displaying Fusion Middleware Control Console"
- Section 6.2, "Navigating to the Home Page for WebCenter Portal"
- Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
- Section 6.4, "Navigating to Dependent Components"

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin`, `Operator`, or `Monitor` role through the Oracle WebLogic Server Administration Console.

See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

6.1 Displaying Fusion Middleware Control Console

System administrators can login to Fusion Middleware Control Console and access pages for managing Oracle WebCenter Portal. Your role determines what you can see and do after logging in. To find out more, see Section 1–7, " WebCenter Portal Operations and Oracle WebLogic Server Roles."

To access the Fusion Middleware Control Console:

1. Start Fusion Middleware Control.

   Fusion Middleware Control is configured for a domain, and it is automatically started when you start the Oracle WebLogic Server Administration Server. See the "Starting and Stopping Fusion Middleware Control" section in *Oracle Fusion Middleware Administrator's Guide*.

2. Enter the following URL in y our browser:
   `http://`*`hostname`*`.`*`domain`*`:`*`port`*`/em`

   For example: `http://myhost.mycompany.com:7001/em`

   The port number is the port number of the Administration Server. By default, the port number is 7001. The port number is listed in `config.xml`:

- On Windows: *DOMAIN_HOME*\config\config.xml
- On UNIX: *DOMAIN_HOME*/config/config.xml

See also, the "About Managing Ports" section in *Oracle Fusion Middleware Administrator's Guide*.

3. Enter a valid administrator **User Name** and **Password** details for the farm.

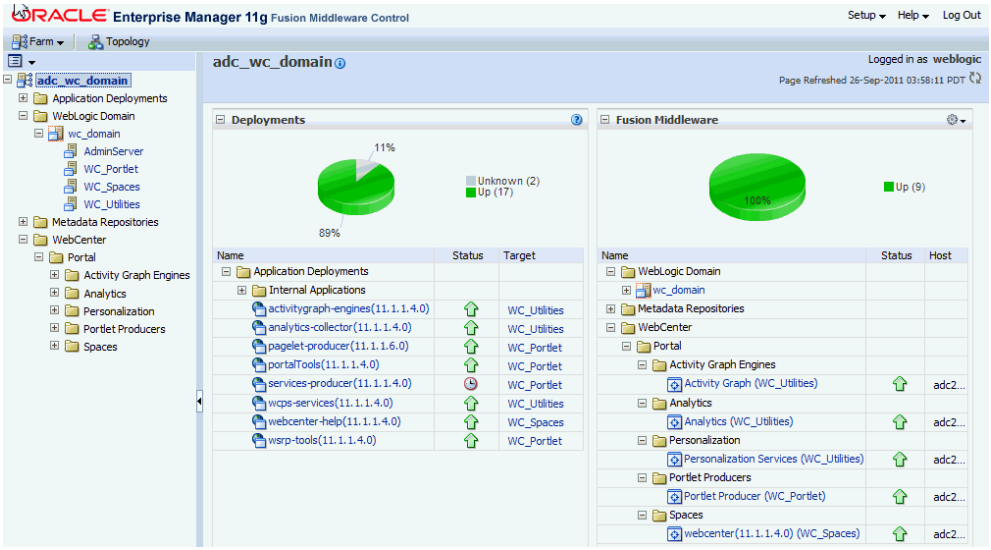
   The default user name for the administrator user is `weblogic`. This is the account you can use to log in to Fusion Middleware Control for the first time.

4. Click **Login**.

   The first page you see is the Farm home page. You can view this page at any time by selecting the name of the farm in the navigation pane (Figure 6–1).

   Tip: If you are unable to log in, try logging in to the WebLogic Administration Console to confirm your host/port/credentials. The Weblogic Admin Console is accessible at the same host/port as Fusion Middleware Control: `http://host.domain:port/console`.

Figure 6–1 Farm Home Page



From the navigation pane, you can drill down to view and manage all components in your farm, including WebCenter Portal and any Portal Framework application that you may have deployed. For detailed instructions, see:

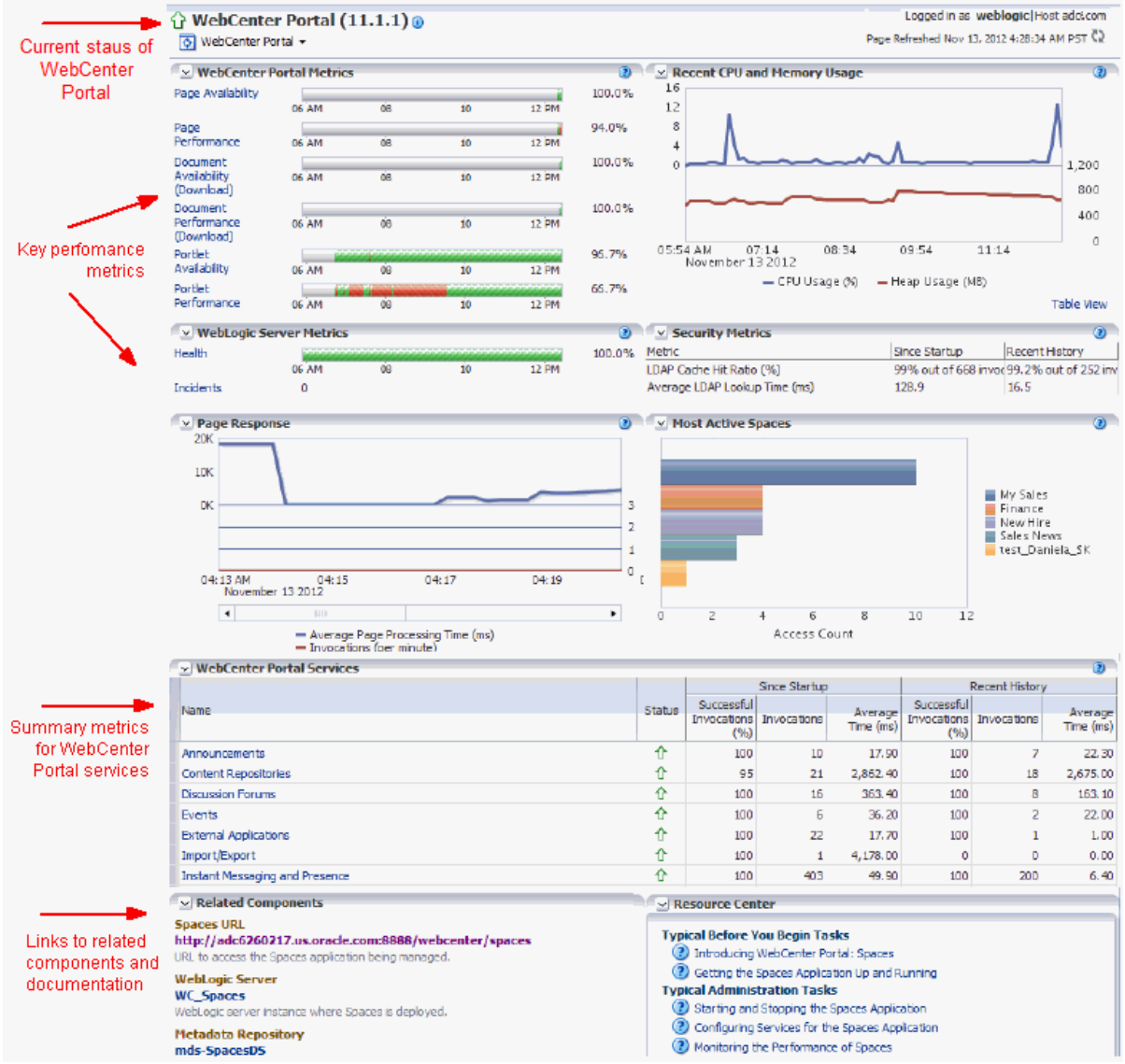
- Section 6.2, "Navigating to the Home Page for WebCenter Portal"
- Section 6.3, "Navigating to the Home Page for Portal Framework Applications"

6.2 Navigating to the Home Page for WebCenter Portal

The WebCenter Portal home page (Figure 6–2) is your starting place for managing Oracle WebCenter Portal's out-of-the-box portal application "*WebCenter Portal*". The page displays status, performance and availability of all the components and tools/services that make up WebCenter Portal.

Note: Versions that display alongside WebCenter Portal product and component names in Fusion Middleware Control do not reflect the true version number of installed products. To verify which WebCenter Portal product version you have installed, refer to Appendix G.2.1, "How Do I Find Out Which Oracle WebCenter Portal Version Is Installed?"

Figure 6–2 WebCenter Portal Home Page



From here you can:

The metrics displayed on WebCenter Portal's home page enable you to:

- Check the status of the WebCenter Portal application and view key performance data.
- Quickly see whether the application is performing as expected through charts that immediately report:

  - availability and performance issues with pages, documents, and portlets
  - general health of the WebLogic Server and the back-end LDAP server

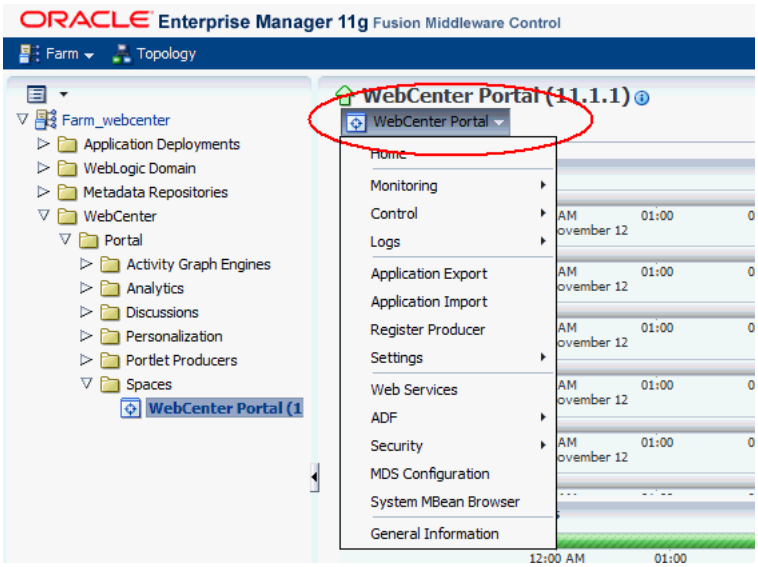
  Hover over the links in the WebCenter Portal Metrics and WebLogic Server Metrics sections for a brief description about the information displayed and click the links to drill down to more detail.

- Monitor CPU and heap memory usage charts to detect whether system resources are running low.
- Track overall response time compared with the user access rate to see how the application preforms under different loads and to diagnose system resource issues.
- Quickly see which portals are used the most, and then drill down to see the slowest performers, and determine which portals are recording the most errors.
- View status and key performance metrics for WebCenter Portal tools/services used in the application.
- Drill down to detailed performance information for individual portals, tools/services, external applications, portlets, and producers.
- Navigate to other key components, including the WebLogic Server managed server on which the WebCenter Portal application is running, and the MDS repository.

Note: To find out more about the performance metrics displayed on the home page, what to look out for, and how to diagnose issues with your installation, see Section 27.1.3, "Using Key Performance Metric Data to Analyze and Diagnose System Health."

The home page for WebCenter Portal also displays a **WebCenter Portal** menu (Figure 6–3).

Figure 6–3 Menu for the WebCenter Portal Application



To navigate to the main home page for WebCenter Portal:

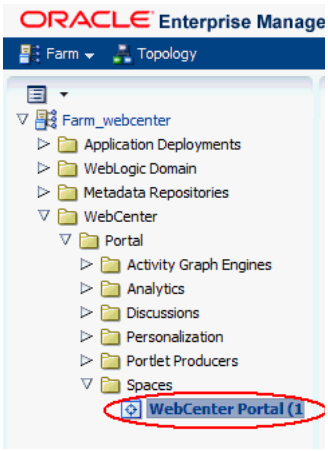
1. Login to Fusion Middleware Control.

   See Section 6.1, "Displaying Fusion Middleware Control Console."

2. In the Navigator (Figure 6–4), expand **WebCenter > Portal > Spaces**.

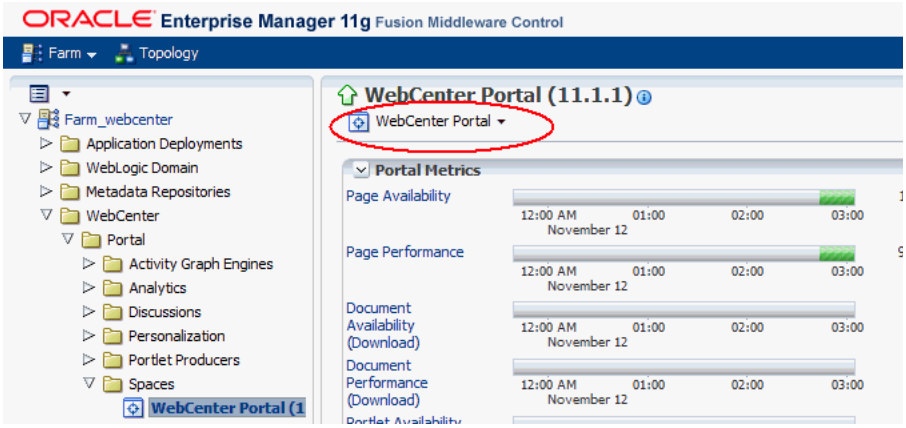
3. Select **WebCenter Portal** to navigate to the home page for your WebCenter Portal installation (Figure 6–4).

Figure 6–4 Navigating to the WebCenter Portal Home Page



Notice how the Navigator menu changes to *WebCenter Portal* (Figure 6–5).

Figure 6–5 Displaying the WebCenter Portal Home Page and Menu



Another way to access the context menu for a particular component is to right-click the node in the navigation tree. For example, if you right-click the **WebCenter Portal (11.1.1)** node (under the **Spaces** node on the left in Figure 6–5), the same *WebCenter Portal* menu displays.

From the **WebCenter Portal** menu, you can:

- Drill down to detailed performance metrics for all components
- Select and chart live metrics
- Start and stop the WebCenter Portal application
- Analyze diagnostic information and configure logs

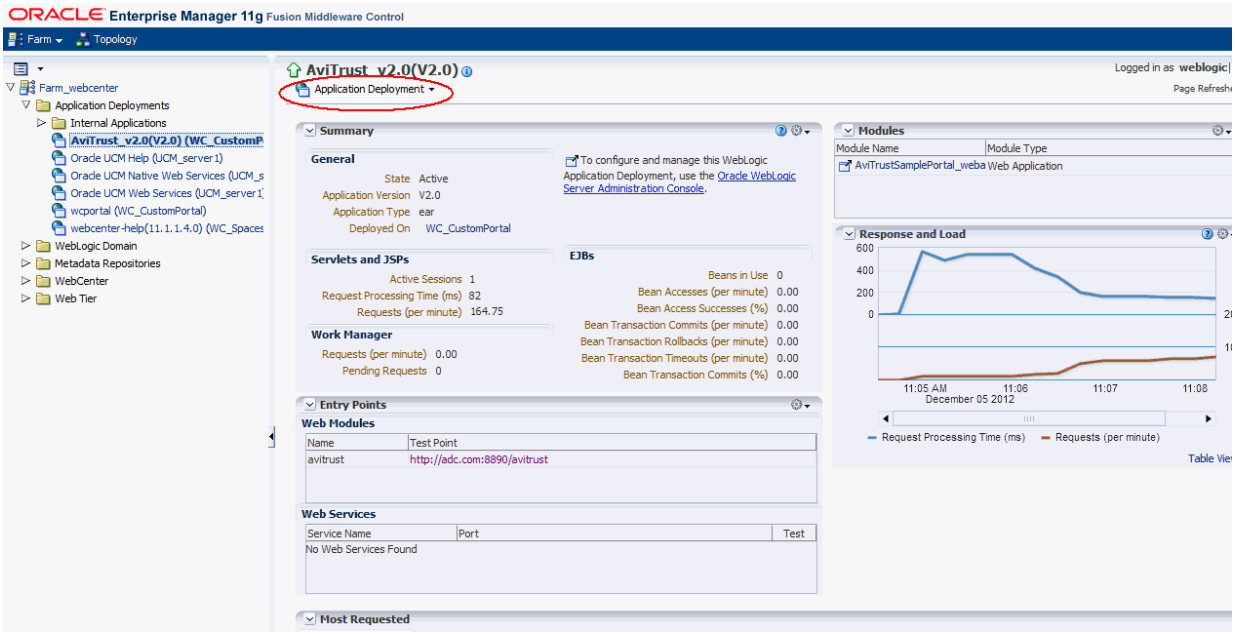
- Export and import the WebCenter Portal application
- Register and manage portlet producers
- Configure application settings
- Manage back-end services
- Manage external applications
- Configure security policies and roles
- Configure ADF and MDS options
- View web services-related information
- Most tasks that you perform from WebCenter Portal's home page are described in this guide. To help guide you to the appropriate chapter or section, refer to Chapter 2, "Getting WebCenter Portal Up and Running."

6.3 Navigating to the Home Page for Portal Framework Applications

The J2EE Application Deployment home page (Figure 6–6) is your starting place for managing portal application deployments developed built using the WebCenter Portal Framework application template in Oracle JDeveloper. The page displays status, performance and availability of all the components and tools/services that make up the Portal Framework application.

Note: Oracle WebCenter Portal's out-of-the-box portal application "*WebCenter Portal*" has a different home page, see Section 6.2, "Navigating to the Home Page for WebCenter Portal."

Figure 6–6 Portal Framework Application Home Page



From here you can:

- Check Portal Framework application status.

- Navigate to the Oracle WebLogic Server Administration Console.
- Access various standard Application Deployment menu options:
  - Start, restart, and shutdown the application
  - View and configure log files
  - Undeploy and redeploy the application
  - Configure security policies and roles
  - Configure ADF and MDS options
- View a performance summary, entry points to the application, Web Services and modules associated with the application, and the response and load data which shows the requests per second and the request processing time.
- Navigate to key components of the Portal Framework application.
- Drill down to detailed performance information for individual modules and tools/services.

> **Note:** To find out more about the performance metrics displayed on the home page, what to look out for, and how to diagnose issues with your installation, see Section 27.1.3, "Using Key Performance Metric Data to Analyze and Diagnose System Health."

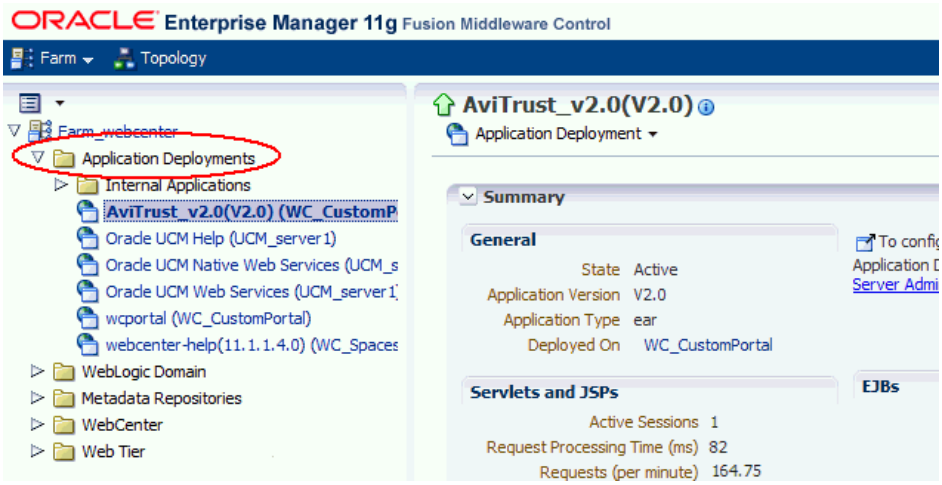
To navigate to the main home page for your Portal Framework application:

1. Login to Fusion Middleware Control.

   See Section 6.1, "Displaying Fusion Middleware Control Console."

2. In the Navigator (Figure 6–7), expand **Application Deployments**.

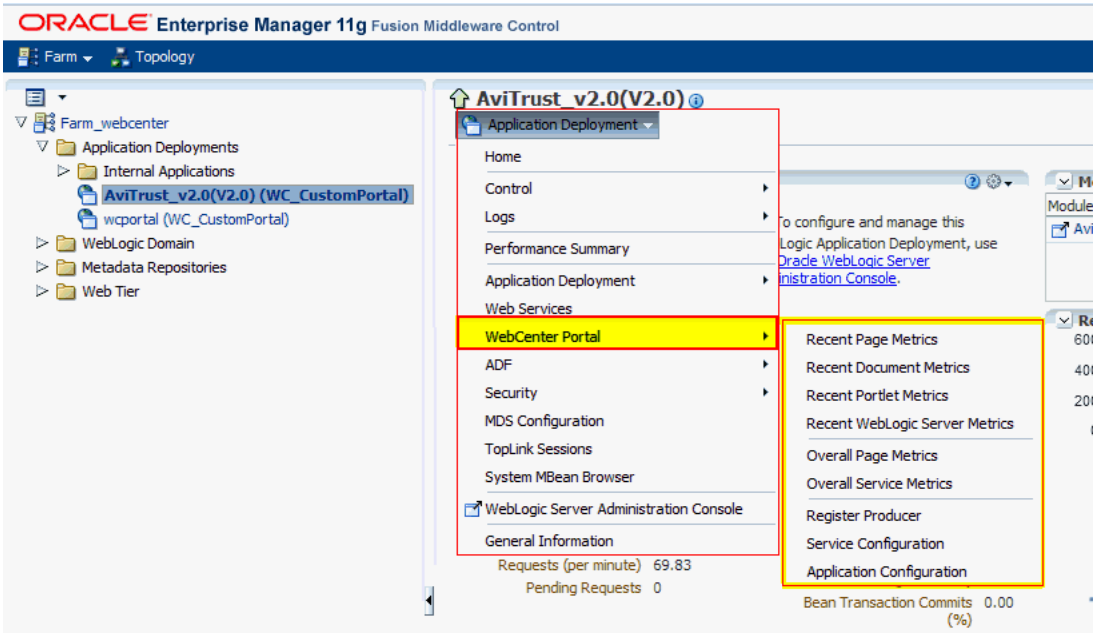
Figure 6–7 Navigating to a Framework Application Home Page



3. Select the name of your Framework application to display the application's home page.

   Notice that the **Application Deployment** menu displays an additional menu option—*WebCenter Portal* (Figure 6–8).

Figure 6–8 Displaying the Portal Framework Application Home Page and Menu



From the WebCenter Portal menu, you can perform WebCenter Portal-specific tasks such as:

- Monitor detailed WebCenter Portal performance metrics:
  - Recent page, document, portlet producer, and WebLogic Server metrics.
  - Overall page and tool/service metrics.

  (See Chapter 27, "Monitoring Oracle WebCenter Portal Performance.")
- Register and manage portlet producers (see Chapter 21, "Managing Portlet Producers").
- Configure application settings, such as which search crawler to use, fine tune various search settings, set up a channel for notifications, and configure a proxy server.
- Manage connections to back-end services (see Chapter 8, "Managing Tools and Services").
- Manage external applications (see Chapter 23, "Managing External Applications").

6.4 Navigating to Dependent Components

From WebCenter Portal pages it is easy to navigate to pages belonging to related components, such as, WebLogic Server domains, servers, Java components, MDS repository, and so on.

- **WebCenter Portal** - From the home page, click links in "Related Components" to navigate to WebCenter Portal application itself, WebLogic Server installation pages, and MDS repository pages in Fusion Middleware Control. See also, Section 6.2, "Navigating to the Home Page for WebCenter Portal."
- **Portal Framework applications** - The Application Deployment menu on the J2EE application home page offers direct navigation to the Oracle WebLogic Server

Administration Console, and pages relating to WebCenter Portal, ADF, MDS repository, and security configuration and administration. See also, Section 6.3, "Navigating to the Home Page for Portal Framework Applications."

7

Starting and Stopping Managed Servers and Applications for Oracle WebCenter Portal

This chapter describes how to restart the managed server on which the application is deployed. Most configuration changes that you make to WebCenter Portal and Portal Framework applications, through Fusion Middleware Control or using WLST, are not dynamic; you must restart the managed server for changes to take effect. For example, when you modify connection details to backend servers, you must restart the application's managed server. There are exceptions; portlet producer and external application registration *is* dynamic. Any new portlet producers and external applications that you register are immediately available in your application and any changes that you make to existing connections take effect immediately too.

This chapter includes the following topics:

You perform all start and stop operations from the Oracle WebLogic Server Administration Console too. See also, the "Starting and Stopping Servers" section in *Oracle Fusion Middleware Managing Server Startup and Shutdown for Oracle WebLogic Server*.

Note: Node Manager *must* be running before you can start and stop administration servers, managed servers, WebCenter Portal and Portal Framework applications through Fusion Middleware Control or Oracle WebLogic Server Administration Console.

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin`, or `Operator` role through the Oracle WebLogic Server Administration Console.

See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

7.1 Starting Node Manager

Node Manager *must* be running before you can start and stop administration servers, managed servers, WebCenter Portal and Portal Framework applications through Fusion Middleware Control or Oracle WebLogic Server Administration Console. Node Manager starts after installation, so you only need to restart Node Manager if someone specifically shuts it down.

For information on how to start Node Manager with `startNodeManager.sh`, see the "Using Node Manager" section in the *Oracle Fusion Middleware Node Manager Administrator's Guide for Oracle WebLogic Server*.

7.2 Starting and Stopping Managed Servers for WebCenter Portal Application Deployments

Most WebCenter Portal configuration changes that you make, through Fusion Middleware Control or using WLST, are not dynamic; you must restart the managed server on which the application is deployed for your changes to take effect.

When you start or restart a managed server, all applications deployed on the managed server start automatically, see also Table 7–1.

Table 7–1 Oracle WebCenter Portal Managed Servers and Applications

Managed Server	Application(s)	
WC_Spaces	webcenter	(WebCenter Portal Application)
	webcenter-help	(WebCenter Portal Online Help)
WC_Portlet	portalTools	(OmniPortlet)
	wsrp-tools	(WSRP Tools)
	pagelet-producer	(Pagelet Producer)
	services-producer	(WebCenter Services Producer)
WC_Collaboration	owc_discussions	(Discussions Server)
WC_Utilities	analytics-collector	(Analytics)
	activitygraph-engines	(Activity Graph)
	wcps-services	(Personalization Services)
WC_CustomPortal	<your_portal_framework_application_name>	

Note: This section describes how to start and stop WebCenter Portal managed servers listed in Table 7–1. To start and stop managed servers for other components, refer to:

- Oracle WebCenter Content managed server, see *Oracle Fusion Middleware Installing and Configuring Oracle WebCenter Content*
- Oracle SOA Server managed server, see *Oracle Fusion Middleware Installation Guide for Oracle SOA Suite and Oracle Business Process Management Suite*

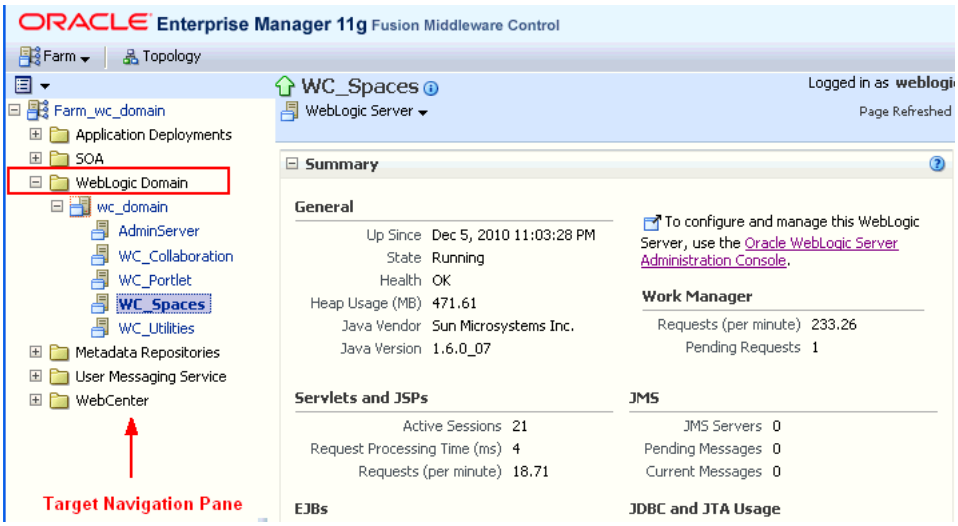
While a specific order in which to start managed servers is not mandated, if you must start multiple managed servers, it is good practice to start the managed server on which WebCenter Portal or your Portal Framework application is deployed last.

To start, stop, or restart a WebCenter Portal managed server through Fusion Middleware Control:

1. Login to Fusion Middleware Control.
2. Expand **WebLogic Domain** in the Target Navigation Pane.
3. Expand **wc_domain**, and select the managed server you want to start or stop.

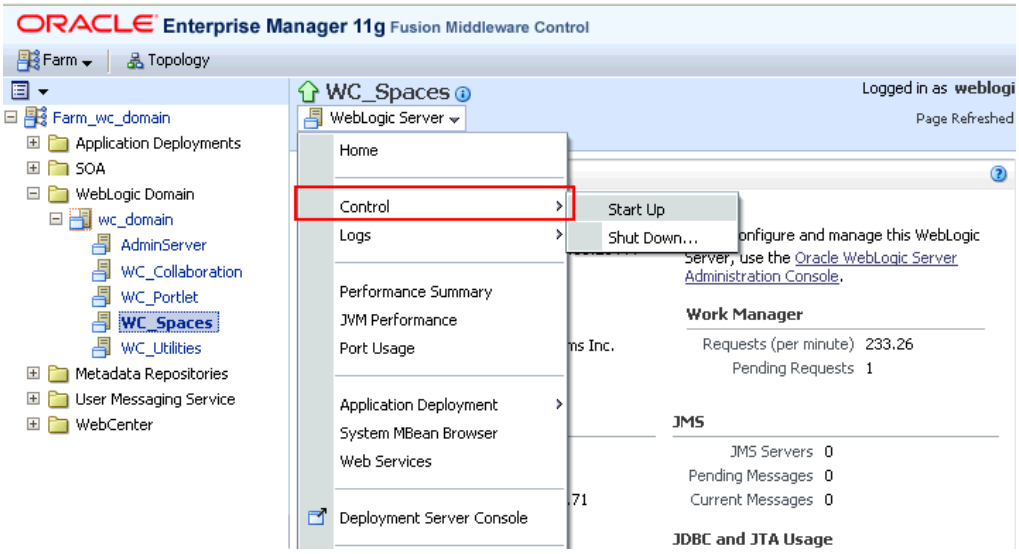
   The home page for the managed server displays (Figure 7–2).

Figure 7–1 Managed Server Home Page



4. From the **WebLogic Server** menu:
   - To start the managed server, select **Control > Start Up**.
   - To stop the managed server, select **Control > Shut Down**.

Figure 7–2 Managed Server Start Up or Shut Down



Alternatively, right-click the name of the managed server in the Target Navigation Pane to access menu options for the managed server.

To start and stop WebCenter Portal managed servers using command line tools, see the "Starting and Stopping Oracle WebLogic Server Instances" section in *Oracle Fusion Middleware Administrator's Guide*.

7.3 Starting and Stopping the WebCenter Portal Application

It's easy to start, restart, and shut down WebCenter Portal from Fusion Middleware Control:

- Starting WebCenter Portal Using Fusion Middleware Control
- Stopping WebCenter Portal Using Fusion Middleware Control

Alternatively, use WLST:

- Starting WebCenter Portal Using WLST
- Stopping WebCenter Portal Using WLST

You can also start WebCenter Portal through Oracle WebLogic Server Administration Console. For information, see the "Displaying the Oracle WebLogic Server Administration Console" section in *Oracle Fusion Middleware Administrator's Guide*.

> **Note:** Application configuration changes require you to restart the *WC_Spaces managed server* on which WebCenter Portal is deployed. For details, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

7.3.1 Starting WebCenter Portal Using Fusion Middleware Control

Starting WebCenter Portal makes the application available to its users; stopping it makes it unavailable.

To start WebCenter Portal through Fusion Middleware Control:

1. In Fusion Middleware Control, navigate to the home page for the WebCenter Portal application.

   See Section 6.2, "Navigating to the Home Page for WebCenter Portal."

2. From the main WebCenter Portal menu, select **WebCenter > Portal> Control> Start Up**.

   Alternatively, right-click **WC_Spaces** in the Target Navigation Pane to access this menu option.

   A progress message displays.

3. Click **Close**.

Note how the application status changes to Up (Green arrow).

7.3.2 Starting WebCenter Portal Using WLST

Use the WLST command `startApplication` to start WebCenter Portal. For command syntax and detailed examples, see the "startApplication" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For the WebCenter Portal application, the `appName` argument is always `webcenter`.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

7.3.3 Stopping WebCenter Portal Using Fusion Middleware Control

When you stop the WebCenter Portal application no one can use it. Stopping an application does not remove its source files from the server; you can later restart a stopped application to make it available again.

When you stop WebCenter Portal, the managed server on which the WebCenter Portal application is deployed (WC_Spaces) remains available.

To stop a WebCenter Portal application through Fusion Middleware Control:

1. In Fusion Middleware Control, navigate to the home page for WebCenter Portal.

   See Section 6.2, "Navigating to the Home Page for WebCenter Portal."

2. From the main menu, select **WebCenter > Portal >Control > Shut Down**.

   Alternatively, right-click **WC_Spaces** in the Target Navigation Pane to access this menu option.

3. Click **OK** to continue.

   A progress message displays.

4. Click **Close**.

Note how the status changes to Down (Red arrow).

7.3.4 Stopping WebCenter Portal Using WLST

Use the WLST command `stopApplication` to stop the WebCenter Portal application. For command syntax and detailed examples, see the "stopApplication" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For the WebCenter Portal application, the `appName` argument is always `webcenter`.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

7.4 Starting and Stopping Portal Framework Applications

It's easy to start and shut down Portal Framework applications from Fusion Middleware Control:

- Starting Portal Framework Applications Using Fusion Middleware Control
- Stopping Portal Framework Applications Using Fusion Middleware Control

Alternatively, use WLST:

- Starting Portal Framework Applications Using WLST
- Stopping Portal Framework Applications Using WLST

You can also start Portal Framework applications through Oracle WebLogic Server Administration Console. For information, see the "Displaying the Oracle WebLogic Server Administration Console" section in *Oracle Fusion Middleware Administrator's Guide*.

Note: Application configuration changes require you to restart the *managed server* on which the Portal Framework application is deployed. For details, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

7.4.1 Starting Portal Framework Applications Using Fusion Middleware Control

Starting a Portal Framework application makes it available to its users; stopping it makes it unavailable.

When you stop a Portal Framework application, the managed server on which it is deployed remains available.

To start a Portal Framework application through Fusion Middleware Control:

1. In Fusion Middleware Control, navigate to the home page for the Portal Framework application.

   See Section 6.3, "Navigating to the Home Page for Portal Framework Applications."

2. From the Application Deployment menu, select **Application Deployment >Control > Start Up**.

   Alternatively, right-click the name of the Portal Framework application in the Target Navigation Pane to access this menu option.

   A progress message displays.

3. Click **Close**.

Note how the application status changes to Up (Green arrow).

7.4.2 Starting Portal Framework Applications Using WLST

Use the WLST command `startApplication` to start a Portal Framework application. For command syntax and detailed examples, see the "startApplication" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

7.4.3 Stopping Portal Framework Applications Using Fusion Middleware Control

When you stop a Portal Framework application no one can use it. Stopping an application does not remove its source files from the server; you can later restart a stopped application to make it available again.

> **Note:** You can also stop WebCenter Portal through Oracle WebLogic Server Administration Console. For information, see the "Displaying the Oracle WebLogic Server Administration Console" section in *Oracle Fusion Middleware Administrator's Guide*.

To stop a Portal Framework application:

1. In Fusion Middleware Control, navigate to the home page for the Portal Framework application.

   See Section 6.3, "Navigating to the Home Page for Portal Framework Applications."

2. From the main menu, select **Application Deployment >Control > Shut Down**.

   Alternatively, right-click the name of the Portal Framework application in the Target Navigation Pane to access this menu option.

3. Click **OK** to continue.

A progress message displays.

4. Click **Close**.

Note how the status changes to Down (Red arrow).

7.4.4 Stopping Portal Framework Applications Using WLST

Use the WLST command `stopApplication` to stop a Portal Framework application. For command syntax and detailed examples, see the "stopApplication" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Part V

Managing Tools, Portlet Producers, and External Applications

This part of *Oracle Fusion Middleware Administering Oracle WebCenter Portal* presents administration tasks for tools, services, portlet producers, and external applications used by WebCenter Portal and Portal Framework applications.

Part V contains the following chapters:

8

Managing Tools and Services

This chapter provides an overview of managing tools and services in WebCenter Portal and Portal Framework applications. It also explains the back-end servers required and provides information on enabling tools and services in portals. These tasks are performed by a system administrator at the application level. Working with tools and services at the portal level is an application specialist or portal moderator task, as described in the "Introduction to Portal Tools and Services" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

This chapter includes the following topics:

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console and the `Administrator` role in the deployed application:

- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.
- **Portal Framework application**: `Administrator` role granted through the Administration Console.

For more information about roles and permissions, see Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

8.1 Introduction to Managing Tools and Services

WebCenter Portal and Portal Framework applications expose collaborative, social networking, and personal productivity features through *tools* and *services*, which, in turn, expose subsets of their features and functionality through *task flows*. Task flows provide reusable functionality that may expose all or a subset of the features available from a particular tool or service.

Some tools, like tags, are available and work out-of-the-box, but other tools require additional configuration (for example, a connection to an *external* back-end server). The following tools and services require a connection to an external data repository or server (such as a content server, a presence server, a discussions server, a mail server) where relevant information is stored:

- Analytics
- Announcements
- Discussions
- Documents, including wikis and blogs
- Events
- Instant Messaging and Presence (IMP)
- Mail
- RSS
- Search (for Oracle SES adapter)
- Worklists

In addition, the following tools and services require a connection to a database schema where relevant information (such as relationship mapping) is stored:

- Activity Graph
- Analytics
- Documents (for documents, wikis and blogs that want to include the comments and Activity Stream)
- Links
- Lists
- People Connections
- Polls
- Tags

Table 8–1 lists where data associated with the various tools and services is stored, that is, in MDS, a database, or an external repository or server. You may find it helpful to know which tools and services are impacted when any one of these repositories are unavailable.

- **MDS** - Some tools and services store connection metadata in the Metadata Services Repository (MDS). Changes that you make to applications, post deployment, are stored in MDS as customizations. For more information, see Section 1.3.5, "Oracle WebCenter Portal Configuration Considerations."

  For WebCenter Portal, MDS is installed and configured out-of-the-box, as described in Section 8.2.1, "Setting Up the MDS Repository." For Portal Framework applications, see Section 42.1.5, "Creating and Registering the Metadata Service Repository."

- **Database** - Some tools and services require a connection to a database schema where relevant information (such as relationship mapping) is stored.

  For more information, see Section 8.2.2, "Setting Up Database Connections."

- **External repository or server** - Some tools and services require a connection to an external data repository (such as a content server, a presence server, or a mail server) where relevant information is stored.

  For more information on setting up those connections, refer to the relevant chapter in this guide (as indicated in Table 8–1). For example, for information on how to set up a connection to an external discussions server, refer to Chapter 12, "Managing Announcements and Discussions" (as indicated in the For More

Information column in Table 8–1).

See Also: Section 8.3, "About Tools and Services in WebCenter Portal" for information specific to WebCenter Portal.

Table 8–1 Data Repositories for Tools and Services

Tools and Services	Description	MDS	Database Schema	External Repository	For More Information
Activity Graph	Recommends people, portals, and content that a user may be interested in connecting with, based on existing connections and shared interaction with objects in the portal		ACTIVITIES schema		Section 8.2.2, "Setting Up Database Connections" Chapter 10, "Managing Activity Graph"
Activity Stream	Provides a streaming view of the activities of your connections, actions taken in portals, and business activities		ACTIVITIES schema		Section 8.2.2, "Setting Up Database Connections" Chapter 16, "Managing People Connections"
Analytics	Enables you to display usage and performance metrics for your portal application		ACTIVITIES schema	X	Section 8.2.2, "Setting Up Database Connections" Chapter 11, "Managing Analytics"
Announcements	Provides the ability to post announcements about important activities and events to all authenticated users	X	DISCUSSIONS schema	X	Section 8.2.1, "Setting Up the MDS Repository" Chapter 12, "Managing Announcements and Discussions"
Discussions	Provides the ability to create threaded discussions, posting and responding to questions and searching for answers	X	DISCUSSIONS schema	X	Section 8.2.1, "Setting Up the MDS Repository" Chapter 12, "Managing Announcements and Discussions"
Documents	Provides content management and storage capabilities, including file upload, file and folder creation and management, file check out, versioning, and so on. Exposes these capabilities through the Documents tool console or task flows such as Document Explorer, Document List Viewer, and Document Manager. Provides components that display an individual file on a page as a linked document, an inline preview, or an image. The documents tool also supports wiki and blog functionality.	X	WEBCENTER schema - for documents (including wikis and blogs) that want to include comments and activity stream	X	Section 8.2.1, "Setting Up the MDS Repository" Section 8.2.2, "Setting Up Database Connections" Chapter 9, "Managing Content Repositories"

Table 8–1 (Cont.) Data Repositories for Tools and Services

Tools and Services	Description	MDS	Database Schema	External Repository	For More Information
Events[1]	Provides the ability to create and maintain a schedule of events relevant to a wider group of authenticated users. Also provides access to your personal events from your Outlook calendar if the Exchange server is configured.	X	WEBCENTER schema (Portal events)[2]	X (Personal Events)	Section 8.2.1, "Setting Up the MDS Repository" Chapter 13, "Managing Calendar Events"
Instant Messaging and Presence (IMP)	Provides the ability to observe the status of other authenticated users (online, offline, busy, or away) and to contact them instantly			X	Chapter 14, "Managing Instant Messaging and Presence" "Using Instant Messaging and Presence Viewer" in *Oracle Fusion Middleware Using Oracle WebCenter Portal*
Links	Provides the ability to view, access, and associate related information; for example, you can link to a document from a discussion		WEBCENTER schema		Section 8.2.2, "Setting Up Database Connections" "Linking Information in WebCenter Portal" in *Oracle Fusion Middleware Using Oracle WebCenter Portal*
Lists	Provides the ability to create, publish, and manage lists	X	WEBCENTER schema		Section 8.2.1, "Setting Up the MDS Repository" Section 8.2.2, "Setting Up Database Connections" "Adding Lists of Information to a Portal" in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*
Mail	Provides easy integration with IMAP and SMTP mail servers to enable users to perform mail functions, such as reading messages, creating messages with attachments, replying to or forwarding messages, and deleting messages	X		X	Section 8.2.1, "Setting Up the MDS Repository" Chapter 15, "Managing Mail"
Messages and Feedback	Provides the ability to post messages, attachments, and feedback for your connections and to the Activity Stream	X	ACTIVITIES schema	X	Section 8.2.1, "Setting Up the MDS Repository" Chapter 16, "Managing People Connections"
Notes[3]	Provides the ability to "jot down" and retain bits of personally relevant information	X			Section 8.2.1, "Setting Up the MDS Repository"

Table 8–1 (Cont.) Data Repositories for Tools and Services

Tools and Services	Description	MDS	Database Schema	External Repository	For More Information
Notifications	Provides a means of subscribing to services and application objects and, when those objects change, receiving notification across one or more messaging channels				Chapter 19, "Managing Subscriptions and Notifications"
People Connections	Provides social networking capabilities, such as creating a personal profile, displaying current status, and viewing other users' recent activities		WEBCENTER schema	X	Section 8.2.2, "Setting Up Database Connections" Chapter 16, "Managing People Connections"
Polls	Enables you to survey your audience (such as their opinions and their experience level), check whether they can recall important information, and gather feedback		WEBCENTER schema		Section 8.2.2, "Setting Up Database Connections"
Profiles	Provides views of users' contact information (such as email address, business address, phone number), department, manager, photo, portal activities, public documents, and connections				Chapter 16, "Managing People Connections"
Recent Activities	Provides a summary view of recent changes to documents, discussions, and announcements	X			Section 8.2.1, "Setting Up the MDS Repository" Chapter 16, "Managing People Connections"
RSS	Provides the ability to access the content of many different web sites from a single location—a news reader	X			Section 8.2.1, "Setting Up the MDS Repository" Section 8.2.4, "Setting Up a Proxy Server" Chapter 17, "Managing RSS"

Table 8–1 (Cont.) Data Repositories for Tools and Services

Tools and Services	Description	MDS	Database Schema	External Repository	For More Information
Search	Provides the ability to search services, the application, or an entire site (This includes integrating Oracle Secure Enterprise Search.)	X		X	Section 8.2.1, "Setting Up the MDS Repository" Chapter 18, "Managing Oracle Secure Enterprise Search in WebCenter Portal"
Tags	Provides the ability to assign one or more personally-relevant keywords to a given page or document	X	WEBCENTER schema		Section 8.2.1, "Setting Up the MDS Repository" Section 8.2.2, "Setting Up Database Connections"
Worklists	Provides a personal view of business processes that require attention	X		X	Section 8.2.1, "Setting Up the MDS Repository" Chapter 20, "Managing Worklists"

[1] Personal and portal events are available in WebCenter Portal; Personal events are available in Portal Framework applications.
[2] Portal events are available in WebCenter Portal only.
[3] Notes is available in WebCenter Portal only; the service is not available in Portal Framework applications.

8.2 Configuring Back-end Data Repositories for Tools and Services

For certain tools and services to work in WebCenter Portal and Portal Framework applications, you must configure various back-end data repositories.

The following sections are included:

- Section 8.2.1, "Setting Up the MDS Repository"
- Section 8.2.2, "Setting Up Database Connections"
- Section 8.2.3, "Setting Up Back-end Server Connections"
- Section 8.2.4, "Setting Up a Proxy Server"
- Section 8.2.5, "Setting Up External Application Connections"
- Section 8.2.6, "Setting Up Composer-Specific Configuration"

8.2.1 Setting Up the MDS Repository

Some tools and services store information in the Metadata Services Repository (MDS). To enable these tools and services in WebCenter Portal or Portal Framework applications, you must configure the MDS repository.

For WebCenter Portal, MDS is installed and configured out-of-the-box.

For Portal Framework applications, see Section 42.1.5, "Creating and Registering the Metadata Service Repository."

> **See Also:** "Managing the Metadata Repository" and "Purging Oracle WebCenter Portal Data" sections in the *Oracle Fusion Middleware Administrator's Guide*.

8.2.2 Setting Up Database Connections

Many tools and services store information in the WebCenter Portal repository, which is a database with the WebCenter Portal schema (`WEBCENTER`) installed. Refer to Table 8–1 for a complete list of these tools and services. For example, with the Links service, relationship mapping information, such as what object is linked to what other object, is stored in this database. Some other tools, such as analytics and activity graph require the ACTIVITIES schema.

For WebCenter Portal, `WEBCENTER` and `ACTIVITIES` schemas are configured out-of-the-box, so no further configuration is required.

For Portal Framework applications, you must set up a database connections, as required. This database connection can be of type **JDBC Data Source** or **JDBC URL**.

See Also:

- "Setting Up a Database Connection" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper* for information on creating the connection and installing the schema
- Section 42.1.6, "Deploying the Application to a WebLogic Managed Server" for data source considerations when deploying your application to a production environment
- Chapter 44, "Managing Export, Import, Backup, and Recovery for Portal Framework Applications" for information on backing up and migrating this information

Depending on the connection type used in an application, do one of the following:

- Create a global data source, if the application does not include an application-level data source with password indirection. For information on creating global data sources, see the "Creating and Managing JDBC Data Sources" section in the *Oracle Fusion Middleware Administrator's Guide*.
- Map the connection credentials, if the application uses an application-level data source with password indirection. The password is set through the Oracle WebLogic Administration Console on the **Credential Mappings** tab under **Security**. If you change the password for an indirect data source on the **Connection Pool** tab under **Configuration**, then it has no effect. For more information on credential mapping, see "JDBC Data Sources: Security: Credential Mapping" under the "Creating a JDBC Data Source" section in *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server*.
- Merge the information stored in the application credential store with that of the global application store, if the application uses a JDBC URL connection. For more information on credential migration behavior, see the "Configuring the Credential Store" section in the *Oracle Fusion Middleware Application Security Guide*.

In a typical business scenario, applications are deployed to different managed servers, and multiple databases are used as repositories for the applications. The repository that you use in a development environment is different from that in a production environment, and therefore, when migrating Portal Framework applications from development to production, you must reconfigure the database connection.

When a repository connection is reconfigured, the local `datasource` file and the `*-jdbc.xml` file in the `WEB-INF` directory of the WAR file are updated with the new connection details. However, the `JNDI Name` and `data source` name remain the same. If you change the `JNDI Name` for any reason, then you must also update the

`adf-config.xml` file. The JNDI name must be of the form `jdbc/`*`connection-name`*`DS`. For example, if the application has a connection name `connection1`, then the JNDI name is `jdbc/connection1DS`.

8.2.3 Setting Up Back-end Server Connections

Some tools and services require a connection to an external data repository (such as a content server, a presence server, or a mail server) where relevant information is stored. Refer to Table 8–1 for a complete list of these tools and services, as well as links to the relevant chapter in this guide where connection configuration is described.

Administrators must always use Fusion Middleware Control or the WLST command-line tool to review and configure back-end server connections for WebCenter Portal and Portal Framework application deployments.

Note: Most changes that you make to services configuration, through Fusion Middleware Control or using WLST, are not dynamic so you must restart the managed server on which the application is deployed for your changes to take effect. For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

8.2.4 Setting Up a Proxy Server

A proxy server is required if you want to enable external RSS news feeds and external links in activity stream task flows in WebCenter Portal or your Portal Framework application. The RSS service and the activity stream service share the same proxy server settings.

You can set up a proxy server using Fusion Middleware Control or WLST.

This section includes the following subsections:

- Section 8.2.4.1, "Setting Up a Proxy Server Using Fusion Middleware Control"
- Section 8.2.4.2, "Setting Up a Proxy Server Using WLST"

8.2.4.1 Setting Up a Proxy Server Using Fusion Middleware Control

To set up a proxy server using Fusion Middleware Control:

1. Log on to Fusion Middleware Control and navigate to the home page for your application:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Application Configuration**.
   - For a Portal Framework application - From the **Application Deployment** menu, select **WebCenter Portal** > **Application Configuration**.

3. In the **Proxy Server** section, enter the host name and the port number of the proxy server. For details, see Table 8–2.

Table 8–2 RSS Proxy Server Details

Field	Description
Proxy Host	Enter the host name of the proxy server.
Proxy Port	Enter the port number on which the proxy server is running.

4. Click **Apply** to save this connection.
5. Restart the managed server to which your application is deployed. For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

8.2.4.2 Setting Up a Proxy Server Using WLST

Use the WLST command `setWebCenterProxyConfig` to specify the proxy host and port number used by RSS news feeds and activity stream task flows. For example:

```
setWebCenterProxyConfig(appName='webcenter', proxyHost='www-proxy.example.com',
proxyPort='80')
```

For command syntax and examples, see the "setWebCenterProxyConfig" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information about how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

> **Note:** To start using new proxy details, you must restart the managed server in which your application is deployed. For more information, see the "Starting and Stopping Managed Servers Using WLST" section in *Oracle Fusion Middleware Administrator's Guide*.

Use the `getWebCenterProxyConfig` command to find out the current proxy host and port used by RSS and activity stream task flows. For example:

```
getWebCenterProxyConfig(appName='webcenter')
```

If you want to delete the current proxy host and port settings, use the `unsetWebCenterProxyConfig` command. For example:

```
unsetWebCenterProxyConfig(appName='webcenter')
```

For more information, see the "Proxy Server" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*

8.2.5 Setting Up External Application Connections

When a tool or service interacts with an application that handles its own authentication, you can associate that application with an external application definition to allow for credential provisioning. For more information about working with external applications, see Chapter 23, "Managing External Applications."

Tip: If you are planning to use the same LDAP server and credentials for some of these tools and services (for example for IMP, Events, and Mail), consider creating a single connection for them, specifying the properties to use across the shared connections.

Creating a shared, single connection is especially useful in cases where the identity store imposes additional restrictions that passwords needs to be changed frequently. If you create only one external application connection, it would help minimize invalid login attempts after password changes, thus preventing chances of password lockout.

The following tools and services permit the use of an external application to connect with and define authentication for it:

- Documents
- Events
- Instant Messaging and Presence
- Mail
- RSS Viewer (when using a secured RSS feed)

8.2.6 Setting Up Composer-Specific Configuration

Composer provides the ability to perform run-time application and user customization in-place in your browser. By default, Composer is configured for various page editor settings such as add-on panel registration and sandbox configuration. If the default values do not suit your requirements, you can modify them for your deployed applications. For more information, see the "adf-config.xml" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

To view or modify Composer configuration, you use the System MBean Browser in Fusion Middleware Control.

To modify the default Composer settings:

1. Log on to Fusion Middleware Control and navigate to the home page for your WebCenter Portal application:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Open the System MBean Browser:
   - For the WebCenter Portal application: From the **WebCenter Portal** menu, select **System MBean Browser**.
   - For Portal Framework applications: From the **Application Deployment** menu, select **System MBean Browser**.
3. For the WebCenter Portal application, navigate to:
   Application Defined MBeans >oracle.adf.share.config >Server: WC_Spaces >Application: webcenter >ADFConfig >ADFConfig (bean) **> ADFConfig > PageEditorConfiguration**

   The Application Defined MBeans page is displayed, as shown Figure 8–1.

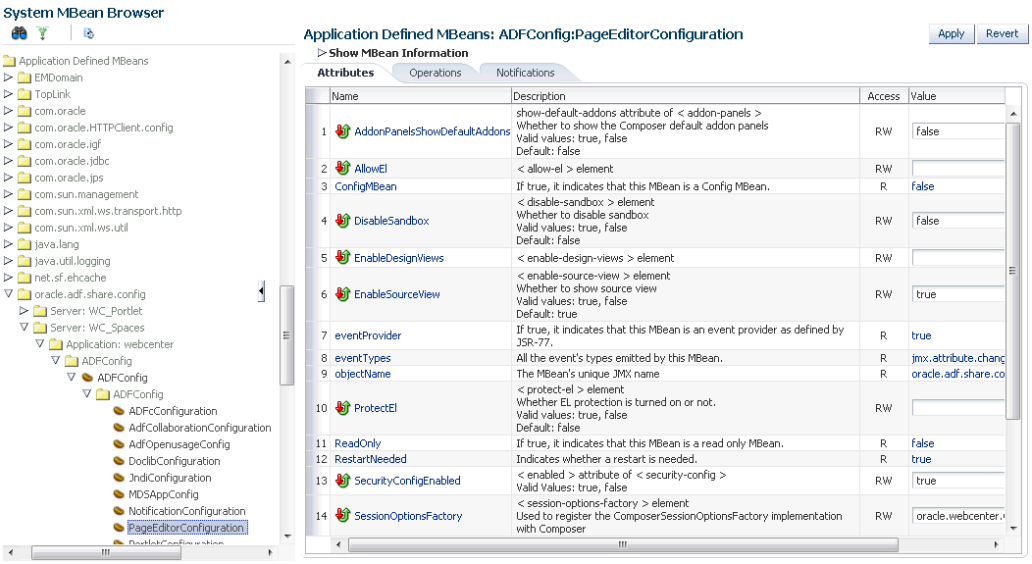
For Portal Framework applications, replace "`Server: WC_Spaces`" with the name of the managed server on which your application is deployed and "`Application: webcenter`" with the name of your application.

Tip: Alternatively, you can navigate to the `PageEditorConfiguration` MBean as follows:

- For WebCenter Portal: **WebCenter Portal Menu > ADF > Configure ADF (adf-config)**
- For Portal Framework applications: **Application Deployment Menu > ADF > Configure ADF (adf-config)**

  Then, navigate to **ADFConfig** (bean) > **ADFConfig** > **PageEditorConfiguration**.

Figure 8–1 System MBean Browser - Composer Properties



4. You can view the description and values for various Composer settings in the Attributes and Operations tab.

   If a setting is not read-only, you can change its value as required. For example, if you want to disable source view, you can set `false` on the `EnableSourceView` attribute. Further, if you want to add or update an add-on panel, you can use the `addOrUpdateAddonPanel` operation under the Operations tab.

   For more information, see the "adf-config.xml" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

5. Click **Apply**.
6. Navigate to the parent `ADFConfig` MBean and select it.
7. In the **Operations** tab, click **Save**.
8. Click **Invoke**.
9. To start using new settings, restart the managed server on which the application is deployed.

For WebCenter Portal, for example, restart the `WC_Spaces` managed server. For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

8.3 About Tools and Services in WebCenter Portal

The system administrator is responsible for managing connections to external servers and also maintains the database schema and Metadata Service (MDS) repositories where application data, specific to WebCenter Portal, is stored. For details, see Chapter 3, "Maintaining WebCenter Portal."

When a back-end server is not configured, intentionally or otherwise, WebCenter Portal cannot offer features or functionality related to that tool:

- Associated task flows are not available in the resource catalog.
- Existing task flows display a message indicating that the tool or service is unavailable.
- Tool or service is not listed as available to moderators—through the portal's administration settings.

When a valid connection exists, the associated tool or service is available in WebCenter Portal. For more information, see Section 8.3.1, "Enabling and Disabling Tools and Services in WebCenter Portal."

The Administration Tools and Services page for WebCenter Portal allows you to perform some optional configurations, if necessary. For more information, see Section 8.3.2, "Configuring Tools and Services for WebCenter Portal."

Reporting Temporary Issues with Tools and Services

When a tool or service is temporarily unavailable, the system administrator can use Fusion Middleware Control to investigate, diagnose, and solve issues relating to services. See also, Section 6.2, "Navigating to the Home Page for WebCenter Portal."

Hiding Task Flows When Tools and Services are Unavailable

Most tools and services are optional. If you decide not to offer a particular tool or service in your application, temporarily or permanently, consider removing any associated task flows that display by default out-of-the-box.

Enabling and Disabling Tools and Services for a Particular Portal

Moderators can enable or disable available tools within their portal. See the "Enabling and Disabling Tools and Services Available to a Portal" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

8.3.1 Enabling and Disabling Tools and Services in WebCenter Portal

WebCenter Portal offers tools and services that allow portal members to collaborate and communicate through various task flows that are associated with these tools and services. Some tools, such as lists, are ready to use out-of-the-box and require no further configuration. Other tools, such as discussions, and other services, such as mail, require connections to the back-end server and require additional configuration. See Section 8.1, "Introduction to Managing Tools and Services."

When a valid connection exists, the associated tool or service is available in WebCenter Portal. With the exception of the Mail service, if the tool or service is not part of a template, then portal moderators or application specialists must enable the tool or

service within a portal. The Mail service is enabled upon portal creation, and, if it is configured by the system administrator, then it cannot be disabled for individual portals. If a tool is included in a portal template, then it is enabled when it is first used. Moderators can manually disable a tool in the portal, with the exception of the Mail service. If mail is configured by the system administrator, then it cannot be disabled for individual portals.

If a moderator manually enables a tool in a portal, WebCenter Portal handles any necessary configuration with the back-end server. For example, when the portal moderator enables discussions in a portal, WebCenter Portal configures discussions storage for that portal on the discussions server and performs role-mapping based authorization, that is, WebCenter Portal roles that allow users to work with the discussions in the portal, are mapped to corresponding roles on the discussions server. See also Section 43.4.2.2.2, "Discussion Server Role Mapping." If role-mapping fails, the portal moderator is notified by email, and users are unable to access discussions.

If a tool is enabled in the template used to create a new portal, WebCenter Portal handles the back-end server configuration when someone accesses that tool for the first time. For example, the first time someone navigates to the Discussions page in a portal at `/webcenter/portal/PortalName/Discussions`, WebCenter Portal configures discussions storage for that portal on the discussions server, performs role-mapping based authorization, and then the discussions page displays.

The following tools and services can be automatically enabled on first use, if the portal template includes it:

- Announcements
- Discussions
- Events
- Lists
- Documents (including wikis and blogs)

> **Note:** In previous releases, these tools and services were enabled at portal creation (instead of on first use). See the "Enabling and Disabling Tools and Services Available to a Portal" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

In most cases, the portal moderators manage tools and services for their own portal, but WebCenter Portal system administrators can also perform this task if required to do so. For more details about enabling and disabling tools and services in a portal, see the "Enabling and Disabling Tools and Services Available to a Portal" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

8.3.2 Configuring Tools and Services for WebCenter Portal

Tools and services are configured by the system administrator by setting up and connecting to the appropriate back-end applications, as described earlier in this chapter.

If the connection exists, the tool or service is available in the portal. Portal moderators are responsible for managing tools and services in their individual portals. You, as the system administrator, can, however, use the Tools and Services page in WebCenter Portal Administration to set up some additional configurations for WebCenter Portal.

To configure options on the Tools and Services page in WebCenter Portal Administration:

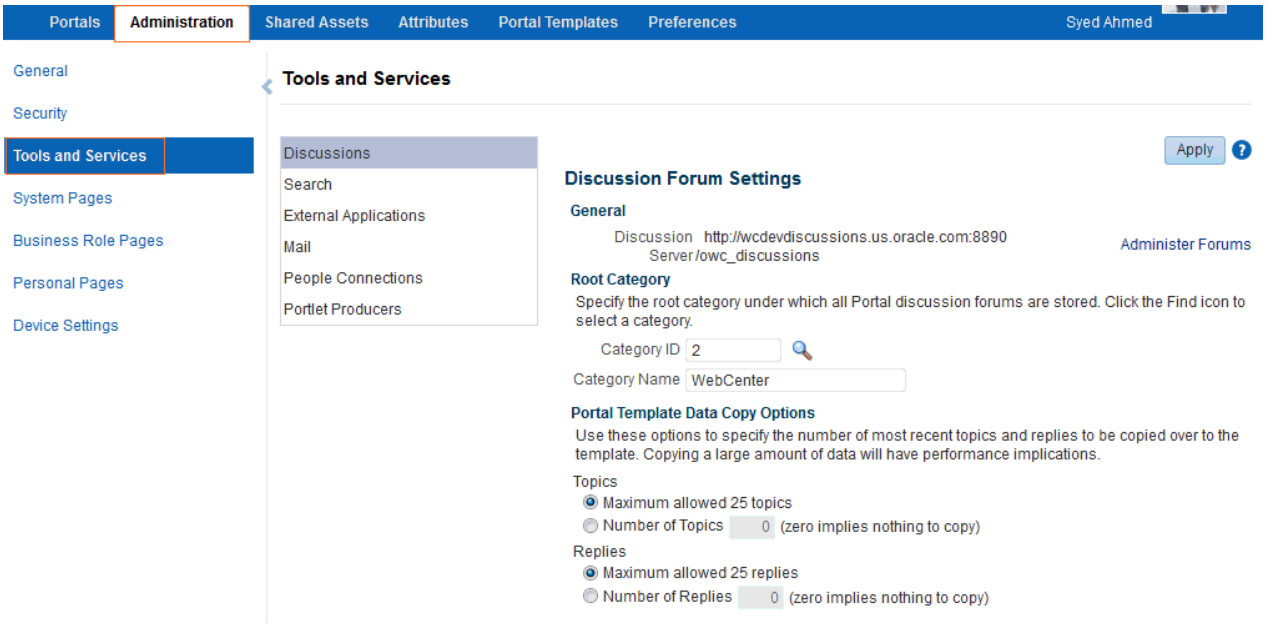
1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Tools and Services**.

   You can also enter the following URL in your browser to navigate directly to the **Tools and Services** page:

   `http://host:port/webcenter/portal/builder/administration/tools`

   See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Figure 8–2 WebCenter Portal Tools and Services Page



2. View the default configurations by clicking the available tabs.

   Click the online Help icon for more information about the tabs.

3. **Optionally**, perform some additional configuration for some of the tools and services.

   - Discussions—change the root category under which discussions are stored. For more information, see Section 12.12, "Configuring Discussion Forum Options for WebCenter Portal."
   - Search—if Oracle SES 11.2.2.2 is configured, you can additionally choose which types of search results to display and do some other customizations. For more information, see Chapter 18, "Managing Oracle Secure Enterprise Search in WebCenter Portal."
   - External Applications— Register new external applications, or edit and deregister the existing external applications. For more information, see Chapter 23, "Managing External Applications."
   - Mail—specify the default mail client for WebCenter Portal, either the local mail client or WebCenter Portal's mail service. For more information, see Section 15.10, "Configuring Send Mail Notifications for WebCenter Portal."

- People Connections—set options for people connection features. For more information, see Section 16.3, "Configuring People Connections for WebCenter Portal."
- Portlet Producers—register new portlet producers, or edit and deregister existing portlet producers. For more information, see Chapter 21, "Managing Portlet Producers."

8.4 About Tools and Services in Portal Framework Applications

When a back-end server is not configured, intentionally or otherwise, the application cannot offer features or functionality related to that tool:

- Associated task flows are not available in the resource catalog.
- Existing task flows display a message indicating that the tool is unavailable.

When a valid connection exists, the associated tool or service is available.

Reporting Temporary Issues with Tools and Services

When a tool or service is temporarily unavailable, system administrators can use Fusion Middleware Control to investigate, diagnose, and solve issues relating to tools and services.

Hiding Task Flows When Tools and Services are Unavailable

Most tools and services are optional. If developers remove or decide not to offer a particular tool or service in a Portal Framework application, temporarily or permanently, consider removing any associated task flows from the application.

9

Managing Content Repositories

This chapter describes how to configure and manage content repositories used by WebCenter Portal and Portal Framework applications.

This chapter contains the following sections:

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console and the `Administrator` role in the deployed application:

- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.
- **Portal Framework application**: `Administrator` role granted through the Administration Console.

For more information about roles and permissions, see Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

9.1 About Content Repositories

Oracle WebCenter Portal's support of the JCR 1.0 open document standard enables integration with multiple back-end content stores. Oracle WebCenter Portal supports the following content repositories: *Oracle WebCenter Content Server* (Content Server), *Microsoft SharePoint*, *Oracle Portal*, and a *file system*

Oracle WebCenter Portal enables content integration through:

- **Content Repository data controls**, which enable read-only access to a content repository, and maintain tight control over the way the content displays in WebCenter Portal and Portal Framework applications.
- **Documents tools**, which enable users to view and manage documents and other types of content in your organization's content repositories.
- **Content Presenter**, which enables end users to select content in a variety of ways and then display those items using available display templates. A Content Presenter task flow can be added during portal development or can be added to editable pages in WebCenter Portal and Portal Framework applications at runtime.

To use these content integration features, at least one connection to a content repository must be configured for WebCenter Portal or your Portal Framework application.

About Content Repository Connections

Portal users need to store, publish, and share files. Documents tools provide content management and storage capabilities for WebCenter Portal and Portal Framework applications, including content upload, file and folder creation and management, file check out, versioning, and so on. To use document tools, you must configure at least one content repository connection and mark it as *active* (also referred to as the *default* content repository):

Note: Both WebCenter Portal and Portal Framework applications support multiple content repository connections.

- **WebCenter Portal** - Any portal (including the Home portal) that enables the documents tool has their own document folder on WebCenter Portal's *default* Content Server. While WebCenter Portal requires the default content repository to be a Content Server, you can also connect WebCenter Portal to any of the other supported repositories.

  For information about setting the default content repository, see Section 9.7, "Changing the Active (or Default) Content Repository Connection". For information about setting additional properties for WebCenter Portal's default content repository, see Section 9.10, "Setting Connection Properties for WebCenter Portal's Default Content Repository".

- **Portal Framework applications** - When a content repository is made active (see Section 9.7, "Changing the Active (or Default) Content Repository Connection"), document task flows use that content repository in instances where no specific connection details are provided. There is no particular requirement on the default content repository used.

When Content Server is the default content repository (mandatory for WebCenter Portal), the Content Server must be connected to the same identity store that is used by WebCenter Portal or your Portal Framework application.

Just like other service connections, postdeployment content repository connections are registered and managed through Fusion Middleware Control or using the WLST command-line tool. Connection information is stored in configuration files and in the MDS repository. For more information, see Section 1.3.5, "Oracle WebCenter Portal Configuration Considerations."

Always use Fusion Middleware Control or the WLST command-line tool to review and configure back-end services for WebCenter Portal and Portal Framework applications. All changes that you make, postdeployment, are stored in the Oracle Metadata Service (MDS) repository as customizations.

> **Note:** Content repository configuration changes that you make through Fusion Middleware Control or using WLST are not dynamic; you need to restart the managed server on which WebCenter Portal or your Portal Framework application is deployed for your changes to take effect. See Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments".

Once connection details are defined, users can expose the content of the connected content repositories through several ADF Faces components, such as `<af:image>`, `<af:inlineFrame>`, and `<af:goLink>`, and built-in Documents task flows (Document Manager, Folder Viewer, and Recent Documents). For more information, see "Working with Web Development Components on a Page" and "Working with the Documents" in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

About Content Repository Configuration Requirements

Prerequisite configuration for the various types of content repository types supported by WebCenter Portal and Portal Framework application are described in the following sections. Before connecting to your repository, ensure that you complete the content repository configuration described here:

- Section 9.2, "Configuring a Content Server Repository"
- Section 9.3, "Configuring a Microsoft SharePoint Repository"
- Section 9.4, "Configuring an Oracle Portal Repository"
- Section 9.5, "Configuring a File System Repository"

Related Information in Other Guides

For more information about managing and including content in WebCenter Portal and Portal Framework applications, see also:

- "Working with Documents" in *Oracle Fusion Middleware Using Oracle WebCenter Portal* to work with documents and document task flows at runtime in WebCenter Portals and Portal Framework applications.

For more information about adding document services to Portal Framework applications and WebCenter Portal using Oracle JDeveloper, see also:

- "Configuring Content Repository Connections" in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper* to configure content repository connections that provide access to decentralized content.
- "Creating Content Presenter Display Templates" in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper* to create custom display templates to integrate and publish decentralized content in WebCenter Portal or Portal Framework applications using Content Presenter.

- "Integrating Documents" in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper* to integrate documents in Portal Framework applications to provide end users with a user-friendly interface to manage, display, and search documents at runtime.

9.2 Configuring a Content Server Repository

This section provides step-by-step instructions for configuring an Oracle WebCenter Content Server 11*g* (Content Server) content repository for WebCenter Portal and Portal Framework applications. Unless otherwise noted, these instructions are common to both WebCenter Portal and Portal Framework applications.

This section contains the following subsections:

- Section 9.2.1, "Prerequisites to Configuring Content Server"
- Section 9.2.2, "Configuration Roadmap for Content Server"
- Section 9.2.3, "Configuring Content Server"

9.2.1 Prerequisites to Configuring Content Server

Read this section to understand the prerequisites and other considerations before continuing with Section 9.2.3, "Configuring Content Server."

This section includes the following subsections:

- Section 9.2.1.1, "Installation Prerequisites"
- Section 9.2.1.2, "Configuration Prerequisites"
- Section 9.2.1.3, "Security Prerequisites"

9.2.1.1 Installation Prerequisites

Content Server

Prior to configuring Content Server 11*g*, you should already have installed Content Server. Content Server is installed as a part of Oracle WebCenter Content, which is an Oracle Fusion Middleware component, and is described in *Oracle Fusion Middleware Installing and Configuring Oracle WebCenter Content*.

If you already have an earlier version of Content Server installed, upgrade your installation to Oracle WebCenter Content Server 11g prior to configuring it. For information about upgrading to Oracle WebCenter Content Server 11g, see the "Upgrading Your Oracle WebCenter Content Environment" chapter in *Oracle Fusion Middleware Upgrade Guide for Oracle WebCenter Content*.

Inbound Refinery

Oracle recommends that you also install Oracle WebCenter Content: Inbound Refinery (Inbound Refinery) as part of the installation. Inbound Refinery is a conversion server that manages file conversions for electronic assets such as documents, digital images, and motion videos. It also provides thumbnail functionality for documents and images and storyboarding for videos. You can use Inbound Refinery to convert content items stored in Content Server. Installing Inbound Refinery is also described in the *Oracle Fusion Middleware Installing and Configuring Oracle WebCenter Content*.

Note: Content Server and Inbound Refinery must be installed in the same domain. Oracle recommends that you install Content Server and Inbound Refinery in the same domain as WebCenter Portal or your Portal Framework application. When they are installed in the same domain, no additional configuration is required to use an external LDAP authentication provider.

9.2.1.2 Configuration Prerequisites

After installing Content Server and Inbound Refinery, you should also have configured the initial post-installation settings described in "Configuring the Content Server Instance" in the *Oracle Fusion Middleware Installing and Configuring Oracle WebCenter Content*. Settings should be configured for both Content Server and Inbound Refinery including the additional WebCenter Portal-specific instructions provided in the tables below. Be sure to restart the servers after updating the settings.

Content Server

Setting	Description
Server Socket Port	This is the intradoc port that we connect to using RIDC (defaults to `4444`). This value is stored in the configuration file (`../config/config.cfg`) for the Content Server Managed Server as `IntradocServerPort`.
Incoming Socket Connection Address Security Filter	Server filter specifying what machines can access Content Server through a socket connection. This value is stored in the configuration file for the Managed Server as `SocketHostAddressSecurityFilter`.
Full Text Search (Optional, but strongly recommended)	Specifies the full-text search engine. "Internal" is the recommended value.

Inbound Refinery

Setting	Description
Server Socket Port	This port is used for communication between Content Server and Inbound Refinery. This value was entered on the post-installation configuration page, and can be found on the Inbound Refinery configuration information page under `Server Port`. You can also find it in the `MW_HOME/user_projects/domains/ucm_domain/ucm/ibr/config/config.cfg` file as `IntradocServerPort`.
Incoming Socket Connection Address Security Filter	Server filter specifying what machines can access Inbound Refinery through RIDC. This value is stored in the configuration file for the Managed Server as `SocketHostAddressSecurityFilter`.
Full Text Search (Optional, but strongly recommended)	Specifies the full-text search engine. "Internal" is the recommended value.

9.2.1.3 Security Prerequisites

Content Server must be configured to use the same identity store LDAP server as WebCenter Portal or your Portal Framework application. For information on how to reassociate the identity store with an external LDAP server, see Section 31.1, "Reassociating the Identity Store with an External LDAP Server."

Content Server and Inbound Refinery must be installed in the same domain. Oracle recommends that you install Content Server and Inbound Refinery in the same domain as WebCenter Portal or your Portal Framework application. When they are installed in the same domain, no additional configuration is required to use an external LDAP authentication provider.

9.2.2 Configuration Roadmap for Content Server

The flow chart in Figure 9–1 provides an overview of the prerequisites and tasks required to get Content Server working in WebCenter Portal and Portal Framework applications. The steps in the flow chart are described in Table 9–1 and the subsections in Section 9.2.3, "Configuring Content Server."

Figure 9–1 Configuring Content Server for WebCenter Portal and Portal Framework Applications

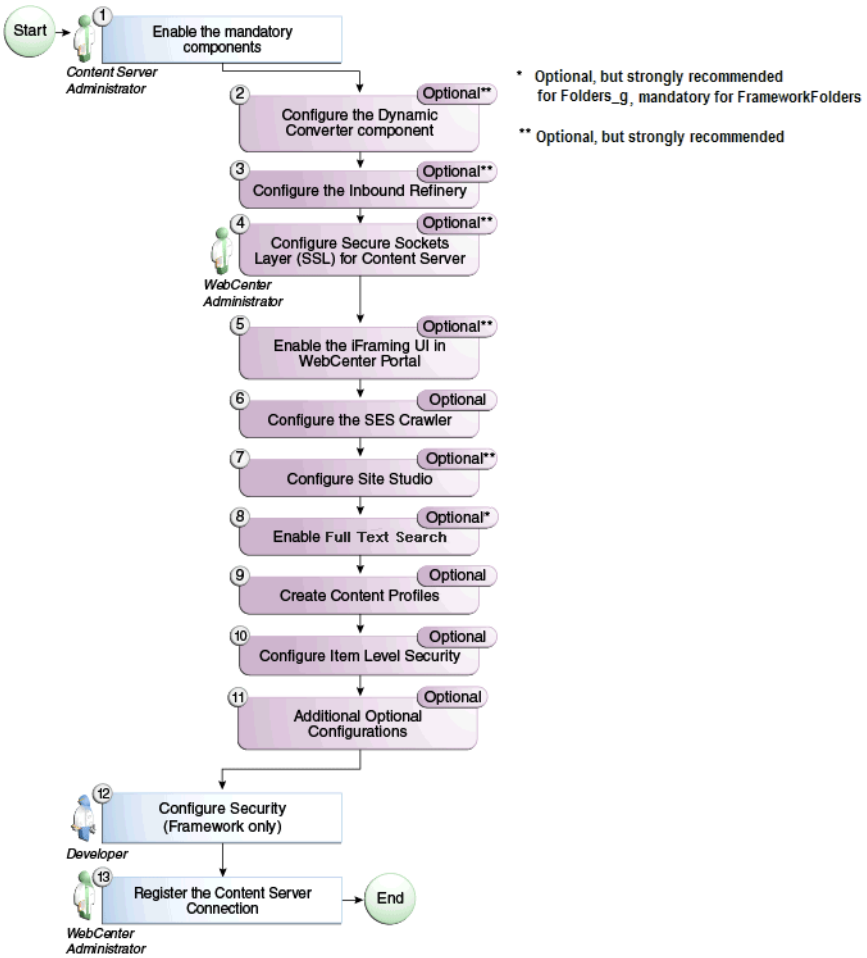


Table 9–1 WebCenter Portal Configuration Tasks for Content Server

Task	Description	Documentation
Enable the mandatory components	**Mandatory** You must enable the WebCenterConfigure component (which configures an instance of Content Server for WebCenter Portal and Portal Framework applications). You must also enable either the Folders_g or the FrameworkFolders component (which provide a hierarchical folder interface to content in Content Server).	See Section 9.2.3.1, "Enabling Mandatory Components."
Configure the Dynamic Converter component	**Optional, but strongly recommended** This component enables HTML renditions. Slide Previewer is available in WebCenter Portal when both DynamicConverter and the WebCenterConfigure components are installed.	See Section 9.2.3.2, "Configuring the Dynamic Converter Component."
Configure the Inbound Refinery	**Optional, but strongly recommended** This is a conversion server that manages file conversions for electronic assets such as documents, digital images, and motion videos. It also provides thumbnail functionality for documents and images and storyboarding for videos. You can use Inbound refinery to convert content items stored in Content Server.	See Section 9.2.3.3, "Configuring the Inbound Refinery."
Configure Secure Sockets Layer (SSL) for Content Server	**Optional, but strongly recommended** To ensure secure identity propagation, you should set up SSL for Content Server.	See Section 35.7, "Securing the WebCenter Portal Connection to Content Server with SSL." Also see Section 9.2.3.4, "Setting Up SSL for Content Server."
Enable the iFraming UI	**Optional, but strongly recommended** If iFraming is not configured, some functionality, such as Document Manager document rendition support, advanced metadata edit, the IFRAME functionality, and so on, will not be available.	See Section 9.2.3.5, "Enabling the iFraming UI." For more information, also see Appendix B, "Oracle HTTP Server Configuration for WebCenter Portal."

Table 9–1 (Cont.) WebCenter Portal Configuration Tasks for Content Server

Task	Description	Documentation
Configure the SES Crawler	**Optional** You can override the default search adapters and use Oracle SES to get unified ranking results for WebCenter Portal resources such as documents, pages, people, and so on.	See Section 18.5.2, "Setting Up Oracle WebCenter Content Server for Oracle SES." Also see Section 9.2.3.6, "Configuring the SES Crawler."
Configure Site Studio	**Optional, but strongly recommended** Configuring Site Studio lets you use Site Studio to create and use Site Studio assets (region definitions and display templates) in Content Presenter. Unless you are absolutely sure you will not need Site Studio, we strongly recommend installing and configuring it so you don't have to come back to it later.	See Section 9.2.3.7, "Setting Up Site Studio." For information, see the "Enabling or Disabling a Component Using the Component Manager" section in *Oracle Fusion Middleware Administering Oracle WebCenter Content* and the "Publishing Content Using Content Presenter" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*. See also *Oracle WebCenter Content Administrator and Manager's Guide for Site Studio*.
Enable a Full-Text Search Option	**For Folders_g: Optional, but strongly recommended** **For FrameworkFolders: Mandatory** Use the OracleTextSearch search option for full-text search. Note that this option should only be used in conjunction with an Oracle database. For MS-SQL, use the DATABASE.FULLTEXT option.	See Section 9.2.3.8, "Enabling Full-Text Search." For more information, also see the "Configuring OracleTextSearch for Content Server" section in *Oracle Fusion Middleware Installing and Configuring Oracle WebCenter Content* and the "Site Studio Integration" section in *Oracle Fusion Middleware Managing Oracle WebCenter Content*.
Create Content Profiles	**Optional** When iFraming is enabled in WebCenter Portal, users have the option to upload content based on Content Server Profiles	See Section 9.2.3.9, "Creating Content Profiles in Content Server." For more information about creating content profiles, see the "Managing Content Profiles" chapter in *Oracle Fusion Middleware Managing Oracle WebCenter Content*.

Table 9–1 (Cont.) WebCenter Portal Configuration Tasks for Content Server

Task	Description	Documentation
Configure Item Level Security	**Optional** Documents tools can use Item Level Security (ILS) to override the default WebCenter Portal document security model, or to expose Content Server document security in a Portal Framework application. Using ILS allows Content Server folders (and their children) or individual documents to have unique security permissions.	See Section 9.2.3.10, "Configuring Item Level Security." See also, "Setting Security Options on a Folder or File" in *Oracle Fusion Middleware Using Oracle WebCenter Portal*
Additional Optional Configurations	**Optional** After completing the rest of your configuration, you can optionally configure the FileStore Provider component and set up Node Manager.	See Section 9.2.3.12, "Additional Optional Configurations."
Configure Security between the Content Server and the Portal Framework application	**Mandatory for Portal Framework applications (not applicable to WebCenter Portal)** To configure Content Server to work with a Portal Framework application, you must first set up content security and users in a development environment and then migrate them to a production environment.	See Section 9.2.3.13, "Configuring Security Between Content Server and WebCenter Portal Framework Applications."
Register the Content Server Repository	**Mandatory** For WebCenter Portal, although in most cases the connection will be configured when WebCenter Portal first starts up, you should at least test it to make sure it has been configured correctly for your environment, and that data has been correctly seeded. For Portal Framework applications, you must configure the connection from the application to Content Server.	For WebCenter Portal, see Section 9.2.3.14.2, "Configuring a Content Server Connection for WebCenter Portal." For WebCenter Portal, be sure to also check the seeded data as described in Section 9.2.3.14.3, "Checking the WebCenter Portal Data Seeded in Content Server." For Portal Framework applications, see Section 9.2.3.14.1, "Configuring a Content Server Connection for Portal Framework applications."

9.2.3 Configuring Content Server

After installing or upgrading to Content Server 11*g*, perform the configuration tasks listed in Table 9–1. Unless otherwise noted, these tasks are common to both WebCenter Portal and Portal Framework applications.

Note: Prior to beginning the configuration you must have completed the installation and configuration steps described in Section 9.2.1, "Prerequisites to Configuring Content Server" that define the starting point for the configuration steps in this section.

Caution: To avoid conflicts and ensure you can migrate documents between multiple Content Server instances, make sure that you've entered a unique Auto Number Prefix for your Content Server instance. To check that the Auto Number Prefix is unique across Content Server instances, log in to Content Server and navigate to **Administration > Admin Server > General Configuration**

This section includes the following subsections:

- Section 9.2.3.1, "Enabling Mandatory Components"
- Section 9.2.3.2, "Configuring the Dynamic Converter Component"
- Section 9.2.3.3, "Configuring the Inbound Refinery"
- Section 9.2.3.4, "Setting Up SSL for Content Server"
- Section 9.2.3.5, "Enabling the iFraming UI"
- Section 9.2.3.6, "Configuring the SES Crawler"
- Section 9.2.3.7, "Setting Up Site Studio"
- Section 9.2.3.8, "Enabling Full-Text Search"
- Section 9.2.3.9, "Creating Content Profiles in Content Server"
- Section 9.2.3.10, "Configuring Item Level Security"
- Section 9.2.3.11, "Showing and Hiding the Wiki Markup Tab in the Rich Text Editor"
- Section 9.2.3.12, "Additional Optional Configurations"
- Section 9.2.3.13, "Configuring Security Between Content Server and WebCenter Portal Framework Applications"
- Section 9.2.3.14, "Registering the Content Server Repository"

9.2.3.1 Enabling Mandatory Components

Mandatory

To prepare Content Server for WebCenter Portal or Portal Framework applications, you must enable either the Folders_g or the FrameworkFolders component, and the WebCenterConfigure component.

This section contains the following subsections:

- Section 9.2.3.1.1, "Considerations for Enabling FrameworkFolders or Folders_g"
- Section 9.2.3.1.2, "Understanding the Folders_g and FrameworkFolders Directory Structure"
- Section 9.2.3.1.3, "Enabling the FrameworkFolders Component"
- Section 9.2.3.1.4, "Enabling the Folders_g Component"

- Section 9.2.3.1.5, "Enabling the WebCenterConfigure Component"

9.2.3.1.1 Considerations for Enabling FrameworkFolders or Folders_g

Oracle WebCenter Content offers two folder solutions: Folders_g and FrameworkFolders. In release 11.1.1.8.3, *new* installations of Oracle WebCenter Portal can be integrated with FrameworkFolders. Previously, Oracle WebCenter Portal only supported Folders_g.

The Folders_g component (or the *Contribution Folders* interface) provides a hierarchical folder interface to content on Content Server. The FrameworkFolders component (or the *Folders* interface) also provides a hierarchical folder interface similar to a conventional file system, for organizing and locating some or all of the content in the repository. However, Folders is a scalable, enterprise solution and is designed to replace Contribution Folders as the folder service for Content Server.

For an Oracle WebCenter Portal instance patched from an earlier release, you *must* continue to use Folders_g. For *new* installations of Oracle WebCenter Portal, it is recommended that you enable the FrameworkFolders component on Content Server for better performance and so as to be able to use any new Content Server features. For information about the criteria that must be met for enabling FrameworkFolders, see the "Preparing Oracle WebCenter Portal for FrameworkFolders Support" section in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*.

9.2.3.1.2 Understanding the Folders_g and FrameworkFolders Directory Structure

Both Folders_g and FrameworkFolders provide a hierarchical folder interface, however, the way content is organized differs in these two setups. This section describes the directory structure used for organizing content in Folders_g and FrameworkFolders.

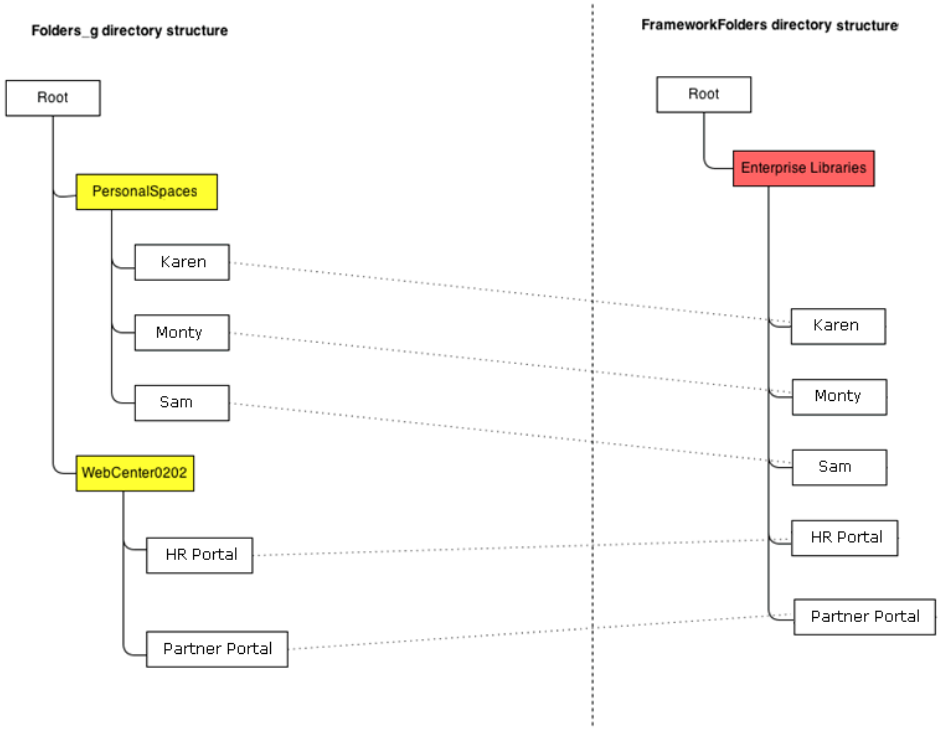
When WebCenter Portal uses a Content Server repository with Folders_g enabled, portals are stored under the WebCenter Portal root folder. Each user has a personal folder in the Home portal that is stored under the path `/PersonalSpaces`, and this folder is named after the user. When WebCenter Portal is configured to use FrameworkFolders, all portal folders and personal folders are treated as enterprise libraries and are stored under the path `/Enterprise Libraries`. Portal folders are named after the portal and personal folders are usually named after the user.

For example, consider a company with the following portals and users:

- `HR Portal`: contains HR policies and documents
- `Partner Portal`: contains documents related to partners
- Users: `Karen`, `Monty`, and `Sam`

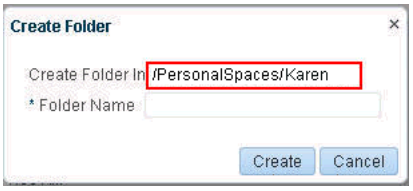
Figure 9–2 shows how WebCenter Portal folders are organized in the Folders_g and FrameworkFolders setups on Content Server. In a Folders_g setup, personal folders for users `Karen`, `Monty`, and `Sam` are organized under `PersonalSpaces`, and the portals `HR Portal` and `Partner Portal` are organized under the WebCenter Portal root folder `WebCenter0202`. In a FrameworkFolders setup, personal folders `Karen`, `Monty`, and `Sam` and the portal folders `HR Portal` and `Partner Portal` are organized under `Enterprise Libraries`.

Figure 9–2 Folders_g and FrameworkFolders Folder Structure on Content Server



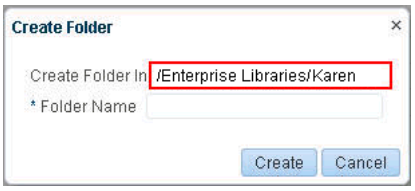
For example, if Karen creates any new folders in the Home portal in a Folders_g setup, the new folders are created under `/PersonalSpaces/Karen` as shown in Figure 9–3.

Figure 9–3 Folder Path When Folders_g is Enabled



If Karen creates any new folders in the Home portal in a FrameworkFolders setup, the new folders are created under `/Enterprise Libraries/Karen` as shown in Figure 9–4.

Figure 9–4 Folder Path When FrameworkFolders is Enabled



9.2.3.1.3 Enabling the FrameworkFolders Component The FrameworkFolders component can be enabled only for new Oracle WebCenter Portal installations that meet the criteria specified in Section 9.2.3.1.1, "Considerations for Enabling FrameworkFolders or Folders_g."

To enable the FrameworkFolders component:

1. Log on to WebCenter Content as an administrator.
2. Choose **Administration**, then **Admin Server**, then **Component Manager** from the **Main** menu.
3. On the Component Manager page, select the **FrameworkFolders** check box.
4. Click **Update**.
5. Click the **advanced component manager** link.
6. On the Advanced Component Manager page, ensure that **Folders_g** appears in the **Disabled Components** list.
7. Restart the Content Server instance.

Note: If Oracle WebCenter Portal is configured to use the FrameworkFolders component, and FrameworkFolders is not enabled, the following message is displayed:

```
Foldering service from content server Folders_g and Portal Server
Configuration FrameworkFolders do not match
```

If you have not applied the WebCenter Content MLR03 patch on release 11.1.1.8.0, the following message is displayed:

```
Framework Folders version on Oracle WebCenter Content Server is not
supported for Oracle WebCenter Portal. The supported versions are
2.1 and later.
```

9.2.3.1.4 Enabling the Folders_g Component

For existing Oracle WebCenter Portal installations patched to the latest release, you must continue to use Folders_g as the folder service on Content Server. Folders_g is incompatible with FrameworkFolders. Therefore, ensure that FrameworkFolders is disabled.

To enable the Folders_g component:

1. Log on to WebCenter Content as an administrator.
2. Choose **Administration**, then **Admin Server**, then **Component Manager** from the **Main** menu.
3. On the Component Manager page, make sure that the **FrameworkFolders** check box is not selected.
4. Click the **advanced component manager** link.
5. On the Advanced Component Manager page, from the **Disabled Components** list box, select **Folders_g** and click **Enable**.
6. Restart the Content Server instance.

Note: If Oracle WebCenter Portal is configured to use the Folders_g component, and Folders_g is not enabled, the following exception displays:

```
SEVERE: UCM feature folders is not installed on server. at
oracle.webcenter.content.integration.spi.ucm.UCMBridge.getBridge(UC
MBridge.java:349) ....
```

9.2.3.1.5 Enabling the WebCenterConfigure Component You must enable the WebCenterConfigure component to configure Content Server for WebCenter Portal and Portal Framework applications. Table 9–2 describes the tasks performed in Content Server when you enable this component.

To enable the WebCenterConfigure component:

1. Log on to WebCenter Content as an administrator.
2. Choose **Administration**, then **Admin Server**, then **Component Manager** from the **Main** menu.
3. On the Component Manager page, select the **WebCenterConfigure** check box.

   Tip: On the Component Manager page, you can choose to select other components like **Dynamic Converter** if you plan to use them as you'll otherwise need to enable them later.

4. Click **Update**.
5. Restart the Content Serverinstance.

Enabling the WebCenterConfigure component performs certain tasks in Content Server. Table 9–2 describes these tasks.

Table 9–2 Tasks Associated with the WebCenterConfigure Component

Tasks	Pointers to Verify the Completion of Tasks
Enables accounts	Content Server > Administration > Admin Server > General Configuration > Enable Accounts checkbox or *MW_HOME*/user_projects/domains/ucm_domain/ucm/cs/config/config.cfg file. The setting in this file is UseAccounts=1.
Allows updates to documents that are yet to be released	Content Server > Administration > Admin Server > General Configuration > Additional Configuration Variables or *MW_HOME*/user_projects/domains/ucm_domain/ucm/cs/config/config.cfg The setting is AllowUpdateForGenwww=1
Disables the cache for folders	If the Folders_g component is enabled, CollectionUseCache is set to false by the WebCenterConfigure component each time the server starts up. This setting is visible in Administration > System Audit Information > Configuration Entry Information > Click All Environment Keys > shows all environment settings. or See the *MW_HOME*/user_projects/domains/ucm_domain/ucm/cs/config/config.cfg file. The setting is CollectionUseCache=1.

Table 9–2 (Cont.) Tasks Associated with the WebCenterConfigure Component

Tasks	Pointers to Verify the Completion of Tasks
Adds metadata fields: ■ `xWCTags` ■ `xWCPageId` ■ `xWCWorkflowAssignment` ■ `xWCWorkflowApproverUserList`	You can view, edit, and add metadata fields here: Content Server > Administration > Admin Applets > Configuration Manager > Information Fields tab.

Sets Folder settings if the `Folders_g` component is enabled:
■ System Default Information Field Configuration: Doc Type = Document
■ Information Field Inherit Configuration
`xWCWorkflowAssignment`
`xWCWorkflowApproverUserList`
	Content Server > Administration > Folder Configuration > System Default Information Field Configuration
	Content Server > Administration > Folder Configuration > Information Field Inherit Configuration

Adds the `WCWorkflowApproverUserToken` workflow token	Content Server > Administration > Admin Applets > Workflow Admin > Options > Tokens menu
Adds three `DynamicConverter` templates	If the `DynamicConverter` component is enabled, the `DynamicConverter` service is called to create the three `DynamicConverter` templates: ■ SLIDE-PREVIEW ■ SLIDE-PREVIEW-TEXT ■ SLIDE-PREVIEW-LARGE
Overrides certain behavior of the Site Studio Switch Content wizard to make Site Studio work in WebCenter Portal and Portal Framework applications	This provides access to the Site Studio Switch Content wizard and the Site Studio Contributor editor from within Content Presenter to allow for adding and editing Site Studio documents from WebCenter Portal or Portal Framework applications. ■ The `contentwizard.hcsp` and `contentwizard.js` files are copied from the `/WebCenterConfigure.zip/component/WebCenterConfigure/publish/contentwizard/` directory to the *`OCS_HOME`*`/cs/weblayout/resources/wcm/custom/sitestudio/contentwizard/webcenter/` directory. ■ The `wcm.sitestudio.form.js` file is copied from the `/WebCenterConfigure.zip/component/WebCenterConfigure/publish/contentwizard/` directory to the *`OCS_HOME`*`/cs/weblayout/resources/wcm/custom/sitestudio/` directory.

Table 9–2 (Cont.) Tasks Associated with the WebCenterConfigure Component

Tasks	Pointers to Verify the Completion of Tasks
Upgrades the PersonalSpace role and default attributes to 11.1.1.6.0 format	If Content Server contains an older version (11.1.1.5.0 and earlier) of the PersonalSpace role format, then enabling WebCenterConfigure upgrades the PersonalSpace role and default attributes to 11.1.1.6.0 format 11.1.1.5.0 and earlier format: Roles: ▪ PersonalSpaceRole with RWD permissions on the PersonalSpaces security group Default Attributes: ▪ All users (public and authenticated) get the PersonalSpaceRole 11.1.1.6.0 format: Roles: ▪ PersonalSpaceRole with R permission on the PersonalSpaces security group ▪ PersonalSpaceAuthenRole with RWD on the PersonalSpaces security group Default Attributes: ▪ All public users get the PersonalSpaceRole ▪ All authenticated users get the PersonalSpaceAuthenRole

9.2.3.2 Configuring the Dynamic Converter Component

Optional, but strongly recommended

This configuration is required for the Slide Previewer capability in WebCenter Portal and Portal Framework applications, which make use of the HTML renditions generated on the fly by the Dynamic Converter.

> **Note:** The Inbound Refinery must also be configured or any previews will fail. See Section 9.2.3.3, "Configuring the Inbound Refinery" for the steps to configure the Inbound Refinery.

The configuration for the Dynamic Converter consists of two steps: enabling the Dynamic Converter, and defining the file types for which the Dynamic Converter is available. If you enabled the Dynamic Converter previously when you were enabling the mandatory components, you can skip the steps to enable it and go directly to the steps for defining the file types.

Enabling the Dynamic Converter

To enable the Dynamic Converter:

1. Log onto the Administration server and open the Admin Server page.

   You can access the Admin Server page through Content Server by going to **Administration > Admin Server**.

2. On the Component Manager page check the **DynamicConverter** checkbox.
3. Click **Update**.
4. Restart the Content Server instance.

Setting the file types to be sent to the Dynamic Converter

To define the file types for which Dynamic Converter is available:

1. Log in to the Content Server and select **Administration > Dynamic Converter Admin > Configuration Settings > Conversion Formats**.

   Note that the Dynamic Converter Admin menu option will not be visible until after you restart the Content Server instance after enabling the Dynamic Converter component.

2. Select the file formats from the dropdown list for which the Dynamic Converter will be enabled. Choose all the document formats for which you want HTML renditions such as Word, Excel, PowerPoint, and PDF.

9.2.3.3 Configuring the Inbound Refinery

Optional, but strongly recommended

The Inbound Refinery is a conversion server that manages file conversions for electronic assets such as documents, digital images, and motion videos. It also provides thumbnail functionality for documents and images and storyboarding for videos. You can use Inbound Refinery to convert content items stored in Content Server. Note that if you enabled the DynamicConverter component (used to generate slide previews), you must also configure the IBR.

To configure Inbound Refinery, you must set up an outgoing provider from Content Server to Inbound Refinery, and specify the file types that will be converted. You also need to enable PDFExportConverter and set other conversion settings on Inbound Refinery. Although optional, you may also want to enable the conversion of wikis and blogs to PDF.

Prior to configuring Inbound Refinery, you should have:

- Installed Inbound Refinery, as described in Section 9.2.1.1, "Installation Prerequisites"
- Completed the initial post-install configuration as described in Section 9.2.1.2, "Configuration Prerequisites"

This section contains the following subsections:

- Section 9.2.3.3.1, "Creating an Outbound Provider"
- Section 9.2.3.3.2, "Enabling PDFExportConverter in Inbound Refinery"
- Section 9.2.3.3.3, "Selecting the File Formats To Be Converted"
- Section 9.2.3.3.4, "Enabling the Conversion of Wikis and Blogs into PDFs"

9.2.3.3.1 Creating an Outbound Provider

Before Content Server can send files to Inbound Refinery for conversion, you must set up an outgoing provider from Content Server to Inbound Refinery with the **Handles Inbound Refinery Conversion Jobs** option checked.

To create an outbound provider:

1. From the Content Server Administration menu, select **Providers**.
2. In the Create a New Provider section of the Providers page, click **Add** in the outgoing row.
3. Enter values for these fields:

- **Provider Name:** Any short name with no spaces describing the Inbound Refinery instance the outgoing provider is for. It is a good idea to use the same name as the Inbound Refinery **Instance Name**.
- **Provider Description:** A description of the outgoing provider.
- **Server Host Name:** The name of the host machine where the Inbound Refinery instance is running (for example, `myhost.example.com`).
- **HTTP Server Address:** The address of the Inbound Refinery instance (for example, `http://myhost.example.com:16250` where `16250` is the Web port).
- **Server Port:** The IntradocServerPort value for the Inbound Refinery instance. This value was entered on the post-installation configuration page, and can be found on the Inbound Refinery configuration information page under **Server Port**. You can also find it in the `MW_HOME/user_projects/domains/ucm_domain/ucm/ibr/config/config.cfg` file as `IntradocServerPort`.

  To display the Inbound Refinery configuration information page:

  – Log in to the Content Server and choose **Administration > Configuration for <instanceName>**.

  – Click **Server Configurations** to display the server configurations.

  Or log into the IBR at **Administration > Admin Server > General Configuration**.
- **Instance Name:** The instance name for Inbound Refinery (the `IDC_Name` value in the `config.cfg` file). This value was entered on the post-installation configuration page as **Server Instance Name**. To find the instance name, log into the Inbound Refinery, and navigate to **Administration -> Configuration for <instanceName>**.
- **Relative Web Root:** The web root of the Inbound Refinery instance (for example, `/ibr/`).

4. Under Conversion Options, check **Handles Inbound Refinery Conversion Jobs**. Do *not* check **Inbound Refinery Read Only Mode**.
5. Click **Add**.
6. Restart Content Server.
7. Go back to the Providers page, and check that the Connection State value is `good` for the provider.

   If the value is not good, double-check that you entered all the preceding entries correctly, and check that the Content Server and Inbound Refinery instances can ping each other.

9.2.3.3.2 Enabling PDFExportConverter in Inbound Refinery

`PDFExportConverter` uses OutsideIn to convert documents directly to PDF files. The conversion can be cross-platform and does not require any third-party product. You can enable `PDFExportConverter` for Inbound Refinery as a server feature.

To enable `PDFExportConverter` on Inbound Refinery:

1. From the Inbound Refinery Administration menu, select **Admin Server** and then **Component Manager**.
2. Select `PDFExportConverter`, and click **Update**.

3. Click **OK** to enable this feature.
4. Restart Inbound Refinery.

To set the PDF converter settings:

1. Log in to Inbound Refinery again.
2. Select **Conversion Settings**, then select **Primary Web Rendition**.
3. Check **Convert to PDF using PDF Export**.
4. Select **Conversion Settings**, then select **Additional Renditions**.
5. Check **Create Thumbnail Images using Outside In**.
6. Select **Conversion Settings > Third Party Application Settings > General OutsideIn Filter Options > Options**.
7. Set the **Path to fonts** to the fonts on the Inbound Refinery system.
8. Select **Use internal graphics rendering** under UNIX Rendering Options.
9. Click **Update**.

For more information, see the "Setting PDF Files as the Primary Web-Viewable Rendition" section in *Oracle Fusion Middleware Administrator's Guide for Conversion*.

9.2.3.3.3 Selecting the File Formats To Be Converted

To tell Content Server which files to send to Inbound Refinery to be converted, you need to select the file formats.

To select the file formats to be converted:

1. From the Content Server Administration menu, select **Refinery Administration** and then **File Formats Wizard**.

   Note: **Refinery Administration** is not listed when there is no valid outgoing provider to an Inbound Refinery instance.

   Content Server displays the File Formats Wizard page. This page configures which file formats will be sent to Inbound Refinery for conversion when they are checked into Content Server.

2. Select the formats you want converted.

   Make sure you check all the file types you want sent to Inbound Refinery for conversion. Do *not* to check HTML, and also do not check **wiki** and **blog** unless you have enabled their conversion through the **WebCenterConversions** component as described in Section 9.2.3.3.4, "Enabling the Conversion of Wikis and Blogs into PDFs."

3. Click **Update**.

9.2.3.3.4 Enabling the Conversion of Wikis and Blogs into PDFs

Optional

Before you can enable the conversion of wikis and blogs into PDFs in WebCenter Portal and Portal Framework applications, you must first:

- Set up the OpenOffice integration with Inbound Refinery. For information, see the "Configuring Inbound Refinery to Use OpenOffice" section in *Oracle Fusion Middleware Managing Oracle WebCenter Content*.
- Perform the steps described in the "Setting Classpath to OpenOffice Class Files" section (see also: "Using OpenOffice Without Logging In to Host") in *Oracle Fusion Middleware Managing Oracle WebCenter Content*.

Enabling the conversion of wikis and blogs into PDFs requires you to first install the WebCenterConversions component, then configure OpenOffice, which converts HTML to PDF, in the Inbound Refinery server and Content Server respectively. The WebCenterConversions component adds the HtmToPDFOpenOffice conversion option, which makes use of OpenOffice conversion in Inbound Refinery (and therefore requires OpenOffice to be configured for that Inbound Refinery).

Note that you must complete the steps below in sequence. If you enable Wiki and Blogs by selecting them in the file Formats Wizard without first installing and enabling the Inbound Refinery the Wiki and Blogs documents will be stuck in the Inbound Refinery conversion queues.

Note: Only images that have been added through the Rich Text Editor (RTE) using the Embed Image feature are visible in the generated PDF. Images referenced with an external URL do not display in the PDF. For information on the RTE, see the "Using the Rich Text Editor (RTE)" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

PDF conversion works only for the new blogs and wikis created after conversion is enabled. Blogs and wikis created before the conversion was enabled cannot be converted to PDFs.

Tip: See also, "File Formats Converted to PDF by Open Office" at *Oracle Fusion Middleware Administrator's Guide for Conversion*.

To install the WebCenterConversion component:

1. Log in to the Inbound Refinery server.
2. Click **Administration** and then select **Admin Server**.

   The Inbound Refinery Admin Server page displays.
3. In the Component Manager, click the **advanced component manager** link.

   The Advanced Component Manager page displays.
4. In the Install New Component section, select **WebCenterConversions.zip** from the Oracle WebCenter Companion CD, then click **Install**.

   The WebCenterConversion component displays in the Disabled Components box.
5. Select **WebCenterConversion** and click **Enable**.
6. Restart the Inbound Refinery server.

To enable the WebCenterConversion component:

1. In the Inbound Refinery server, under **Conversion Settings**, click the **Conversion Listing** link.

   This displays the Conversion Listing page.

2. In the **Conversions** table, select the **Accept** checkbox for `HtmToPDFOpenOffice`, and click **Update**.

The Wiki and Blog options will now appear in Content Server's File Formats Wizard in the associated Content Server instance.

To enable Wiki and Blogs to be converted to PDFs in Content Server:

1. Log in to Content Server.
2. Expand the **Administration** node, then **Refinery Administration**, and then click **File Formats Wizard**.
3. Under **Select File Types**, select the **Wiki** and **Blogs** checkboxes and click **Update**.

To enable the PDF conversion in Inbound Refinery:

1. Log in to the Inbound Refinery server again.
2. Select **Conversion Settings**, and then select **Primary Web Rendition**.
3. Check the **Convert to PDF using Open Office** option.
4. Click **Update**.

9.2.3.4 Setting Up SSL for Content Server

If WebCenter Portal (or a Portal Framework application) and the Content Server you intend to create a repository connection to are not on the same system or the same trusted private network, then identity propagation is not secure. To ensure secure identity propagation you must also configure SSL for Content Server. For a step-by-step description of how to set up SSL for Content server, see Section 35.7, "Securing the WebCenter Portal Connection to Content Server with SSL."

9.2.3.5 Enabling the iFraming UI

Optional, but strongly recommended

WebCenter Portal and Portal Framework applications use Content Server UI presented in an iFrame for certain functionality, such as Document Manager document rendition and advanced metadata editing. iFrame does not support cross-domain communications, so if WebCenter Portal (or Portal Framework application) and Content Server are not in the same domain (in terms of their web address) you must configure the Oracle HTTP Server (OHS), as described below, or iFraming functionality not be available.

> **Note:** Before enabling support for iFraming, you should already have installed and configured the Oracle HTTP Server (OHS) as described Section 33.2.5, "Installing and Configuring the Oracle HTTP Server."

To enable the iFraming UI in WebCenter Portal and Portal Framework applications:

1. Open the `mod_wl_ohs.conf` file and make sure it points to the right Content Server instance.

   The default location of this file is:
   `OHS_HOME``/`*`Oracle_WT1`*`/instances/`*`instance1`*`/config/OHS/ohs1/mod_wl_ohs.conf`

2. Update the connection property of the Content Server to:

```
webContextRoot='/cs'
```

Note that this setting should never be set if OHS is not set up or it is not working correctly.

3. If there is more than one Content Server, reconfigure the second one to use a different context root.
4. Configure OHS by updating the `mod_wl_ohs.conf` file with the Content Server and adfAuthentication protected URI information. For example:

```
<Location /cs>
SetHandler weblogic-handler
WeblogicHost example.com
WeblogicPort 9400
</Location>

<Location /adfAuthentication>
SetHandler weblogic-handler
WeblogicHost example.com
WeblogicPort 9400
</Location>
```

If your Content Server is configured with the Oracle AutoVue VueLink servlet, include the additional entry:

```
<Location /vuelink>
SetHandler weblogic-handler
WeblogicHost example.com  # Same as /cs entry
WeblogicPort 9400    # Same as /cs entry
</Location>
```

For more information about configuring OHS through the `mod_wl_ohs.conf` file, see Appendix B, "Oracle HTTP Server Configuration for WebCenter Portal."

5. Log in to WebCenter Portal or your Portal Framework application and check that the iFraming functionality is available.

Note that since WebCenter Portal (or your Portal Framework application) is now front-ended by OHS, when you access WebCenter Portal or the Portal Framework application you need to do so through OHS. Consequently, you would access your application using the following URL:

```
http://host:OHSPort/appname
```

For example:

```
http://example.com:7777/webcenter
```

9.2.3.6 Configuring the SES Crawler

Optional

Follow the steps in Section 18.5.2, "Setting Up Oracle WebCenter Content Server for Oracle SES" to configure the SES crawler.

9.2.3.7 Setting Up Site Studio

Optional, but strongly recommended

Although configuring Site Studio is strictly speaking optional, without it you will not be able to create and use Site Studio-related assets in Content Presenter. Unless you are

absolutely sure you will not need Site Studio, we strongly recommend installing and configuring it now rather than having to come back to it later.

To enable Site Studio:

1. Log in to Content Server and open the Admin Server Page.

   The Component Manager Page displays.

2. Click **All Features**.

   All components from the Document Management, Folders, Inbound Refinery, Integration, and Web Content Management categories are displayed.

3. Select the checkbox for each component you want to enable. The following components should be enabled:

   - LinkManager
   - SiteStudio
   - DBSearchContainsOpSupport (required for Full Text Search)

4. Click **Update**.

5. Restart the Content Server instance.

6. Log back into Content Server and open the Administration page.

7. Select Site Studio Administration, and then Set Default Project Document Information.

8. Accept the defaults and click **Update**.

9. Select **Site Studio Administration**, and then **Set Default Web Asset Document Information**.

10. Accept the defaults and click **Update**.

11. Finally, configure the cookie path to the context root of your application to prevent losing your session when editing Site Studio content every time the iFrame closes. To do this add a session descriptor to the `weblogic.xml` file (in the WEB-INF folder) and specify the cookie path (which is the context root of your application):

```
<session-descriptor>
  <cookie-path>/app_name</cookie-path>
</session-descriptor>
```

12. To use the Site Studio Designer, log into the Content Server console and navigate to `my downloads`, download Site Studio Designer and install it.

9.2.3.8 Enabling Full-Text Search

Folders_g setup: Optional, but strongly recommended

FrameworkFolders setup: Mandatory

Oracle recommends that you implement full-text search using the OracleTextSearch option. By default, the database used by Content Server is set up to provide metadata-only searching and indexing capabilities. However, you can modify the default configuration of the database to additionally support full-text searching and indexing.

Note that this option should only be used in conjunction with an Oracle database; the OracleTextSearch index must always be in an Oracle database, regardless of the database type used for the main schema. For more information, see the "Configuring

OracleTextSearch for Content Server" section in *Oracle Fusion Middleware Installing and Configuring Oracle WebCenter Content*, and the "Site Studio Integration" section in *Oracle Fusion Middleware Managing Oracle WebCenter Content*.

9.2.3.9 Creating Content Profiles in Content Server

Optional

When iFraming is enabled in WebCenter Portal or your Portal Framework application, users have the option to upload content using Content Server Profiles. For more information on Content Server Profiles, see the "Managing Content Profiles" section in *Oracle Fusion Middleware Managing Oracle WebCenter Content*.

In addition to the mandatory fields needed to upload files to Content Server, for the upload profiles to work correctly in Document Library and WebCenter Portal, the Content Server profiles should also contain the following fields:

- **`xCollectionID`** - for the folder name to be persisted
- **`xIdcProfile`** - for the profile value to be persisted
- **`dRevLabel`** - required by the `CHECKIN_SEL_FORM` API to enable a new version to be checked in

These fields can be added as hidden fields to the profile.

9.2.3.10 Configuring Item Level Security

Optional

Document tools can use Item Level Security (ILS) to override the default document security model in WebCenter Portal, or to expose Content Server document security in a Portal Framework application. Using ILS allows Content Server folders (and their children) or individual documents to have unique security permissions.

This section includes the following sections:

- Section 9.2.3.10.1, "About Item Level Security."
- Section 9.2.3.10.2, "Configuring Item Level Security"
- Section 9.2.3.10.3, "Configuring Additional Settings for WebCenter Portal Framework Applications"

9.2.3.10.1 About Item Level Security

WebCenter Portal and Portal Framework applications allow custom permissions to be set on a file or a folder. This feature is referred to as Item level Security (ILS). Once configured, the feature can be accessed in your application, for example, from the Documents page of a portal.

Note: In WebCenter Portal, using ILS as the primary security mechanism for a portal may become difficult to administer when the number of users grow. Moreover, ILS may not be as efficient as the WebCenter Portal security model. Therefore, Oracle recommends using ILS only to define security for the documents or folders that do not fit within the WebCenter Portal security model (for example, documents and folders to which only a restricted set of users have access). For information about security, see the "Managing Roles and Permissions for a Portal" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

ILS can be used to replace the existing file or folder security with a custom set of permissions.

- When applied to a file, the custom permissions affect only that file.
- When applied to a folder, the updated security is propagated to all child files and folders recursively, stopping when a folder is encountered with its own custom permissions. The propagation does not affect a file with its own custom permissions, if already set.

> **Note:** In WebCenter Portal, ILS cannot be applied to the root folder of a portal. This is so that the portal's security can be correctly restored on a file or folder when its item level security is removed.

Within the Content Server, ILS is implemented as a combination of ACL, account, and other metadata field settings. Content Server must be correctly configured to enable ILS. See, Section 9.2.3.10, "Configuring Item Level Security" and Section 9.2.3.13, "Configuring Security Between Content Server and WebCenter Portal Framework Applications."

What Happens in Content Server on Setting Custom Permissions

The following occurs in Content Server on setting custom permissions for a file or folder from the Item Level Security dialog:

- The account is changed to account `WCILS/original_account`.

  All `AUTHENTICATED` users are by default granted `RWDA` on account `WCILS`, and all `PUBLIC` users are granted `R` on the account `WCILS`. Changing the account to `WCILS/original_account` ensures that only the custom permissions determine the security on the content.
- The ACL content metadata fields, `xClbraUserList` and `xClbraRoleList` are updated with the custom permissions (`xClbraUserList` contains the permissions a user has on a document or folder, and `xClbraRoleList` contains the permissions a group has on the document or folder.)
- The content metadata field, `xInhibitUpdate` is set to `true`, to prevent ILS from overwriting an item's own custom security with a parent folder's custom permissions.

What Happens in Content Server on Removing Custom Permissions

Removing custom permissions from a folder or file attempts to revert the security on that item to the security set on the item's parent folder. When you remove custom permissions, the following changes take place within Content Server:

- The item's account is changed to be the account of its parent folder.
- The item's ACL content metadata fields, `xClbraUserList` and `xClbraRoleList` are cleared.
- The content metadata field, `xInhibitUpdate` is set to `false`.

These changes are propagated in the same way as when the item level security is set.

Prerequisites for Using Item Level Security in WebCenter Portal and WebCenter Portal Framework Applications

For WebCenter Portal or Portal Framework applications, the Item Level Security (ILS) feature is supported only if the application's Content Server security configuration

meets certain prerequisites. In most scenarios ILS is not required, and therefore, it should not be enabled unless explicitly needed. Typical reasons for using ILS are application situations when the Content Server security models need to be overridden or supplemented to handle exception cases to security policies for individual users or groups of users, on a per document basis. Please be aware that there are performance impacts and additional administrative overhead when using ILS.

> **Note:** Oracle recommends using Content Server security because it is efficient and scales easily for a large number of users and content objects compared with item level security. From an administrative perspective, Content Server's security is also easier to maintain. For information about configuring security, see Section 9.2.3.13, "Configuring Security Between Content Server and WebCenter Portal Framework Applications."

The following are Content Server security ILS prerequisites for WebCenter Portal or a Portal Framework application:

- Security is based on Content Server *Accounts* alone.

  Since all content must also have a security group, this means all application users must have `RWD` permissions granted to the application's security group. This is necessary because of how ILS works, that is, on setting the custom permissions, the account automatically changes to `WCILS/original_account`, which is an account all users have `RWDA` granted to. This is so that the custom permissions alone determine the security on the document or folder.

- The content metadata field, `xForceFolderSecurity` is set to `true` for the entire application content. That is, `Folder` security settings are enforced on child folders and documents. This is necessary to support the propagation of custom permissions.

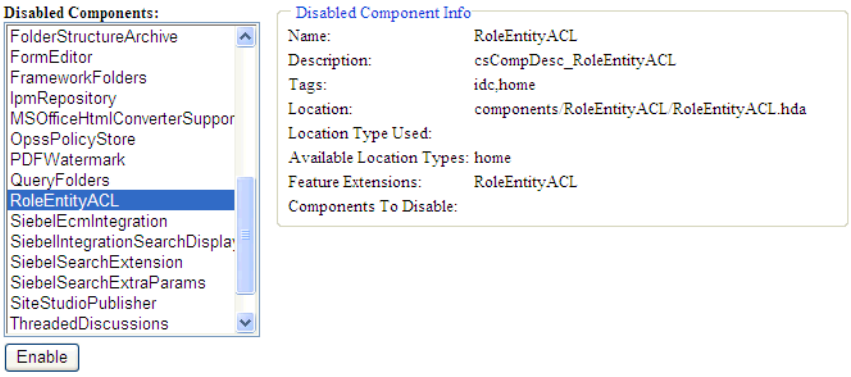
9.2.3.10.2 Configuring Item Level Security

To configure Item Level Security (ILS):

1. Log on to your Content Server instance.
2. From the Administration menu, select **Admin Server** to open **Component Manager**.
3. In the **Component Manager** section, click the **Advanced Component Manager** link.
4. In the Advanced Component Manager page, scroll down to the **Disabled Components** list, select **RoleEntityACL**, as shown in Figure 9–5, and then click **Enable**.

> **See Also:** "Setting Security Options for a File" in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

Figure 9–5 Advanced Component Manager - RoleEntityACL Component



5. From the **Options** pane on left, select **General Configuration**.
6. Under the General Configuration page, in the **Additional Configuration Variables** box, add the following parameters:

   ```
   UseEntitySecurity=1
   SpecialAuthGroups=PersonalSpaces,securityGroup
   ```

   where:

   `SpecialAuthGroups` is a comma separated list (no spaces allowed between values) of security groups. The ILS option is enabled only on content in these security groups.

   - For Portal Framework applications, the *`securityGroup`* is the name of the security group in which content is created.
   - For WebCenter Portal, the name of the security group that contains the WebCenter Portal data is the same as application name you configured to identify WebCenter Portal in the Content Server. You can find this application name using either Fusion Middleware Control or WLST.

     In Fusion Middleware Control, the Application Name property displays in the Add/Edit Content Repository Connection page for the default Content Server connection for WebCenter Portal.

     Using WLST, you can display the application name using the `listDocumentsSpacesProperties` command. For example:

     ```
     listDocumentsSpacesProperties(appName='webcenter')

     The Documents Spaces container is "/myspacesroot"
     The Documents repository administrator is "weblogic"
     The Documents application name is "myWebCenterPortalApp" <- applicationName
     The Documents primary connection is "myContentServer"
     ```

7. Restart Content Server and the managed server on which WebCenter Portal or your Portal Framework application is running.

9.2.3.10.3 Configuring Additional Settings for WebCenter Portal Framework Applications

For a Portal Framework application, in addition to the steps described in Section 9.2.3.10.2, "Configuring Item Level Security", ensure that all users by default are granted `RWDA` on the `WCILS` account. To do this, use the SET_DEFAULT_ATTRIBUTES service. For information about this service, see the

"SET_DEFAULT_ATTRIBUTES" section in *Oracle Fusion Middleware Services Reference for Oracle WebCenter Content*.

To run the SET_DEFAULT_ATTRIBUTES service through a browser:

1. From a browser, log into Content Server as an administrative user.
2. View the source for the page, and find the value of the idcToken by searching for a line containing `var idcToken =` (for example, `var idcToken = 1316188662243:6FE5F809A3B122277B7A1D19912FBB5`).
3. While in the same browser window, enter the URL in the format:

   ```
   http://host:port/cs/idcplg?IdcService=SET_DEFAULT_ATTRIBUTES&dECPropSubKey=<Sec
   urity Group>&dDefAttribs=account,WCILS,15&idcToken=<idcToken>&IsSoap=1
   ```

   For example:

   ```
   http://myhost.com:4444/cs/idcplg?IdcService=SET_DEFAULT_ATTRIBUTES&dECPropSubKe
   y=Custom&dDefAttribs=account,WCILS,15&idcToken=1291297336399:6E324367FC9D2F8BE5
   25F4CEBF4463FC&IsSoap=1
   ```

9.2.3.11 Showing and Hiding the Wiki Markup Tab in the Rich Text Editor

Optional

When creating or editing a wiki document in the Rich Text Editor (RTE), the **Wiki Markup** tab is hidden by default. To show and hide the **Wiki Markup** tab, you can edit the configuration file `blog-wiki-config.xml.xml`.

WARNING: Switching between the Wiki Markup tab and other tabs in the RTE may cause data loss. For this reason, the Wiki Markup tab is disabled by default. Before you enable the Wiki Markup tab, consider potential issues that may result.

For a WebCenter Portal application

To show and hide the **Wiki Markup** tab for portals:

1. Export the latest configuration file `blog-wiki-config.xml.xml` from MDS:

   ```
   exportMetadata(application='webcenter', server='WC_Spaces',
   toLocation='/scratch/aime1',
   docs='/oracle/webcenter/doclib/config/mdssys/cust/site/webcenter/blog-wiki-conf
   ig.xml.xml')
   ```

2. If the configuration file is not found, create it at the path specified in Step 1, then edit the file to add the following code:

   ```
   <?xml version='1.0' encoding='UTF-8'?>
   <mds:customization version="11.1.1.64.86"
   xmlns:mds="http://xmlns.oracle.com/mds" motype_local_name="adf-blogwiki-config"
   motype_nsuri="http://xmlns.oracle.com/webcenter/blogwiki/config">
   <mds:modify
   element="(xmlns(mds_ns1=http://xmlns.oracle.com/webcenter/blogwiki/config))/mds
   _ns1:adf-blogwiki-config/mds_ns1:properties/mds_ns1:property[@name='wiki.markup
   .enabled']">
   <mds:attribute name="value" value="false"/>
   </mds:modify>
   </mds:customization>
   ```

3. Edit the configuration file to change the value of element `wiki.markup.enabled`:

```
<mds:modify
element="(xmlns(mds_ns1=http://xmlns.oracle.com/webcenter/blogwiki/config))/mds
_ns1:adf-blogwiki-config/mds_ns1:properties/mds_ns1:property[@name='wiki.markup
.enabled']">
<mds:attribute name="value" value="true"/>
</mds:modify>
```

   where:

   - `true`: show the **Wiki Markup** tab
   - `false` (default): hide the **Wiki Markup** tab

4. Import the updated file to MDS:

```
importMetadata(application='webcenter', server='WC_Spaces',
fromLocation='/scratch/aime1',
docs='/oracle/webcenter/doclib/config/mdssys/cust/site/webcenter/blog-wiki-conf
ig.xml.xml')
```

For a Portal Framework application

To show and hide the Wiki Markup tab for a Portal Framework application:

1. Export the latest configuration file `blog-wiki-config.xml.xml` from MDS:

```
exportMetadata(application='webcenter', server='WC_Spaces',
toLocation='/scratch/aime1',
docs='/oracle/webcenter/doclib/config/mdssys/cust/site/webcenter/blog-wiki-conf
ig.xml.xml')
```

2. If the configuration file is not found, create it at the path specified in Step 1, then edit the file to add the following code:

```
<?xml version='1.0' encoding='UTF-8'?>
<mds:customization version="11.1.1.64.86"
xmlns:mds="http://xmlns.oracle.com/mds" motype_local_name="adf-blogwiki-config"
motype_nsuri="http://xmlns.oracle.com/webcenter/blogwiki/config">
<mds:modify
element="(xmlns(mds_ns1=http://xmlns.oracle.com/webcenter/blogwiki/config))/mds
_ns1:adf-blogwiki-config/mds_ns1:properties/mds_ns1:property[@name='wiki.markup
.enabled']">
<mds:attribute name="value" value="false"/>
</mds:modify>
</mds:customization>
```

3. Edit the configuration file to change the value of element `wiki.markup.enabled`:

```
<mds:modify
element="(xmlns(mds_ns1=http://xmlns.oracle.com/webcenter/blogwiki/config))/mds
_ns1:adf-blogwiki-config/mds_ns1:properties/mds_ns1:property[@name='wiki.markup
.enabled']">
<mds:attribute name="value" value="true"/>
</mds:modify>
```

   where:

   - `true`: show the **Wiki Markup** tab
   - `false` (default): hide the **Wiki Markup** tab

4. Import the updated file to MDS:

```
importMetadata(application='myPortalFrameworkApp', server='WC_CustomPortal',
fromLocation='/scratch/aime1',
docs='/oracle/webcenter/doclib/config/mdssys/cust/site/site/blog-wiki-config.xml.xml')
```

9.2.3.12 Additional Optional Configurations

This section describes additional optional configurations that are not required for Content Server to function correctly, but nonetheless offer value and comprise best practices for a Content Server enterprise installation.

This section includes the following subsections:

- Section 9.2.3.12.1, "Configuring the File Store Provider"
- Section 9.2.3.12.2, "Setting Up Node Manager"

9.2.3.12.1 Configuring the File Store Provider

A file store for data management is used in the Content Server system instead of the traditional file system for storing and organizing content. The File Store Provider component is installed, enabled, and upgraded by default for a new Content Server instance (with no documents in it). The File Store Provider component automatically upgrades the default file store (DefaultFileStore) to make use of functionality exposed by the component, including modifying the web, vault, and web URL path expressions.

The File Store Provider component exposes the file store functionality in the Content Server interface and allows additional configuration options. For example, you can configure the Content Server instance to use binary large object (BLOB) data types to store content in a database, instead of using a file system.

With File Store Provider, checked-in content and associated metadata are examined and assigned a storage rule based on criteria established by a system administrator. Criteria can include metadata, profiles, or other considerations. The storage rule determines how vault and web files are stored by the Content Server system and how they are accessed by a web server.

The File Store Provider component enables you to define data-driven rules to store and access content managed by the Content Server system. The configuration steps below create a storage rule that ensures content is stored in the database rather than on the file system.

To create a storage rule:

1. Log in to the Content Server instance as system administrator.
2. Select **Administration**, then **Providers**.

   The Providers Page displays.
3. Click **Info** in the Action column next to the `DefaultFileStore` provider.

   The File Store Provider Information Page displays.
4. Specify a name for the rule (for example, `DBStorage`) and select JDBC Storage.
5. Click **OK**.

   The Edit File Store Provider Page displays.
6. Click **Update**.

7. Restart the Content Server instance.

9.2.3.12.2 Setting Up Node Manager

As an additional step to configuring and managing Content Server and the other servers in the domain in which it resides, you may want to consider using Oracle WebLogic Server Node Manager. Node Manager lets you start and stop WebLogic Server instances remotely, monitor them, and automatically restart them after an unexpected failure. You can configure Content Server, the Administration Server, and Node Manager to work together in a WebLogic Server domain. Node Manager is installed on all the machines that host any server instance. For more information about using Node Manager, see the "Using Node Manager with Oracle WebCenter Content" section in the *Oracle Fusion Middleware Installing and Configuring Oracle WebCenter Content*.

9.2.3.13 Configuring Security Between Content Server and WebCenter Portal Framework Applications

Mandatory for Portal Framework applications

To configure Content Server to work with a Portal Framework application, you must first set up content security and users in a development environment and then migrate them to a production environment. For detailed information about security, see also the "Advanced Administration: Security" part in *Oracle Fusion Middleware Administering Oracle WebCenter Content*.

This section describes the following mandatory steps:

- **Creating security groups:** All content items, whether that be a folder or a document, must reside in a security group. Security groups are required for folders so the folder content can be restricted or its access can be customized based on who should view, edit, or manage the folder content. To create security groups follow the steps in Section 9.2.3.13.1, "Creating a Security Group Using the Content Server Console."
- **Creating roles:** Roles are created with different permissions such as, read, write, delete, administer, and are used to define permissions on security groups. First, create the roles in Content Server, as described in Section 9.2.3.13.2, "Creating Roles Using the Content Server Console," and then for the Portal Framework application, as described in Section 9.2.3.13.3, "Creating Roles (Groups) for the Portal Framework application."
- **Creating folders:** Folders include content such as files, subfolders, images. To create folders, follow the steps in Section 9.2.3.13.4, "Creating a Folder Using the Content Server Console."
- **Creating users:** Users are assigned different roles based on their roles and responsibilities in their organizations. Create users as described in Section 9.2.3.13.5, "Creating Users for an External LDAP" or Section 9.2.3.13.6, "Creating Users for the Embedded LDAP," and then grant roles to these users, as described in Section 9.2.3.13.7, "Granting a Role to an External LDAP User" or Section 9.2.3.13.8, "Granting a Role to an Embedded LDAP User."
- **Migrating security:** Migrate these security groups, folders, users, and roles to your production environment. For information, see Section 9.2.3.13.9, "Migrating Security to a Production Environment."
- **Checking the configuration:** check that the security groups and roles have been created correctly as described in Section 9.2.3.13.10, "Checking Your Security Group and Roles Configuration."

The procedures described in this section apply to the Documents service (including wikis and blogs) and Content Presenter.

9.2.3.13.1 Creating a Security Group Using the Content Server Console

To create a security group:

1. Log into the Content Server Console as an administrator.
2. From the Administration menu, select **Admin Applets**.
3. On the Administration Applet page, click **User Admin** to display the User Admin dialog.
4. From the Security menu, select **Permissions by Group**.
5. In the Permission By Group dialog, click **Add Group**.
6. In the Add New Group dialog, enter a group name, for example, `WikiBlog`.
7. Click **OK**.

The security group, which you will use when you create a folder in Section 9.2.3.13.4, "Creating a Folder Using the Content Server Console," is created.

9.2.3.13.2 Creating Roles Using the Content Server Console

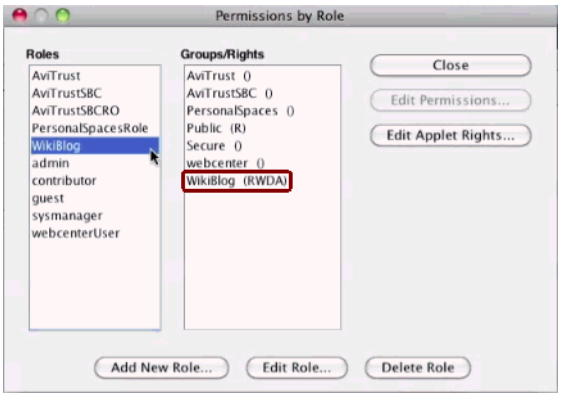
This section describes how to set up two roles in Content Server that mimic those you'll set up in the Portal Framework application: one granting only read permission to the security group, and another granting all permissions to the security group.

To create roles:

1. Log into the Content Server Console as an administrator.
2. From the Administration menu, select **Admin Applets**.
3. On the Administration Applet page, click **User Admin** to display the User Admin dialog.
4. Create a new role with full access:
   a. From the Security menu, select **Permissions by Role**.
   b. In the Permission By Group dialog, click **Add New Role**.
   c. In the Add New Role dialog, enter a name, for example, `WikiBlog`.
   d. Click **OK**. This displays the Permission By Role dialog.
   e. In the Groups/Rights column, select the security group that you created earlier (for example, `WikiBlog`), as described in Section 9.2.3.13.1, "Creating a Security Group Using the Content Server Console."
   f. Click **Edit Permissions**.
   g. In the Edit Permissions dialog, select all checkboxes: Read, Write, Delete, and Admin, and click **OK**.

      RWDA access is enabled, as shown in Figure 9–6.

Figure 9–6 RWDA Permissions



5. Create another role (for example `WikiBlogRO`) with only Read access following steps **4a** to **4f** and selecting the **Read** checkbox in the Edit Permissions dialog in step **4g**.

9.2.3.13.3 Creating Roles (Groups) for the Portal Framework application

This section steps you through creating two roles in the Portal Framework application: one role with read access, and another with full access (read, write, delete, administer).

To create roles (groups):

1. Log into Fusion Middleware Control as an administrator.
2. Under **Domain Structure**, click **Security Realms**.
3. In the table under the Summary of Security Realms section, click **myrealm**, for example.

   IMPORTANT: **myrealm** uses the embedded LDAP that ships with Oracle WebCenter Portal. If your installation uses a different LDAP, such as OID, you must select that instead of the one used by the embedded LDAP.
4. Select the **Users and Groups** tab and then the **Groups** subtab.
5. Under the **Groups** section, click **New** to display the Create a New Group section.
6. In the **Name** field, enter the name of the role to which you granted full access in Content Server (for example, `WikiBlog`), as described in Section 9.2.3.13.2, "Creating Roles Using the Content Server Console," and click **OK**.
7. Create a role or group with the read permission (for example, `WikiBlogRO`) by performing steps 5 and 6. The name of this role must match that you specified in Content Server, as described in Section 9.2.3.13.2, "Creating Roles Using the Content Server Console."

9.2.3.13.4 Creating a Folder Using the Content Server Console

To create a folder:

1. Log into the Content Server Console as an administrator.
2. From the Browse Content menu, select **Contribution Folders** to display the root directory in which you will create a folder.
3. On the Contribution Folders page, from the New Item menu, select **New Folder** to display the Hierarchy Folder Configuration page.

4. In the **Virtual Folder Name** field, enter a meaningful name (for example `WikiBlog`).
5. Under the Folder Information section, in the **Title** field, enter a meaningful title (for example, `WikiBlog`).
6. From the Security Group dropdown, select the security group that you created as described in Section 9.2.3.13.1, "Creating a Security Group Using the Content Server Console."

   All items in this folder will inherit the security from this security group.
7. Click **Save**.

9.2.3.13.5 Creating Users for an External LDAP

This section steps you through creating a user using an LDIF file that you can add to an external LDAP such as Oracle Internet Directory (OID). Note that OID users are more typically managed using ODSM (described in the section on "Managing Directory Entries" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory*).

To create a user:

1. Create an LDIF file as based on the following example:

```
dn: uid=john.doe,ou=people,ou=oidrealm,dc=wc_domain
description: John Doe
cn: john.doe
uid: john.doe
sn: Doe
objectclass: wlsUser
objectclass: organizationalperson
objectclass: inetOrgPerson
objectclass: person
objectclass: top
userpassword: MyPassword
displayName: John Doe
employeeNumber: 12345
employeeType: Regular
givenName: John
homePhone: 650-555-1212
mail: john.doe@example.com
title: Manager
manager: uid=mary.jones,ou=people,ou=myrealm,dc=wc_domain
preferredLanguage: en
departmentNumber: tools
facsimiletelephonenumber: 650-555-1200
mobile: 650-500-1200
pager: 650-400-1200
telephoneNumber: 650-506-1212
postaladdress: 200 Oracle Parkway
l: Redwood Shores
homepostaladdress: 123 Main St., Anytown 12345
```

2. Run the following `ldapadd` command to add the user to OID:

```
ldapadd -D <user_dn> -w <password> -h <myhost.mycompany.com> -p <port> -vf
<file_name.ldif>
```

   For example:

```
ldapadd -D cn=john.doe -w MyPassword -h abcd123.example.com -p 1234 -vf
```

```
newusers.ldif
```

9.2.3.13.6 Creating Users for the Embedded LDAP

This section steps you through creating two embedded LDAP users: a user for the read role, and a role for the full access (read, write, delete, administer) role.

To create users:

1. Log into Fusion Middleware Control as an administrator.
2. Under Domain Structure, click **Security Realms**.
3. In the table under the Summary of Security Realms section, click **myrealm**, the built-in realm that works with the embedded LDAP.
4. Select the **Users and Groups** tab and then the **Users** subtab.
5. Under the **Users** section, click **New** to display the Create a New User section.
6. In the **Name** field, specify a name, for example `Joe`.
7. In the **Password** field, specify a password.
8. In the **Confirm Password** field, enter the password again, and then click **OK**.
9. Create another user by performing steps 4 to 8.

9.2.3.13.7 Granting a Role to an External LDAP User

To grant a role to a user:

1. Create an LDIF file based on the following example containing groups with users assigned to each group:

```
dn: cn=WikiBlog,cn=Groups,dc=us,dc=example,dc=com
objectclass: wlsUser
objectclass: organizationalperson
objectclass: inetOrgPerson
objectclass: person
objectclass: top
cn: WikiBlog
description: Group of WikiBlog
displayname:WikiBlog group
uniquemember: cn=john.doe,cn=users,dc=example,dc=com
```

2. Run the following `ldapadd` command to add the user to OID.

```
ldapadd -D <user_dn> -w <password> -h <myhost.mycompany.com> -p <port> -vf
<file_name.ldif>
```

   For example:

```
ldapadd -D cn=john.doe -w MyPassword -h abc123.example.com -p 1234 -vf
addnewusers.ldif
```

   Note that no restart is required for the managed servers.

9.2.3.13.8 Granting a Role to an Embedded LDAP User

This section steps you through granting the roles you created in Section 9.2.3.13.3, "Creating Roles (Groups) for the Portal Framework application" to the users you created in Section 9.2.3.13.6, "Creating Users for the Embedded LDAP".

To grant a role to a user:

1. Log into Fusion Middleware Control as an administrator.
2. Under Domain Structure, click **Security Realms**.
3. In the table under the Summary of Security Realms section, click **myrealm**, the built-in realm that works with the embedded LDAP.
4. Select the **Users and Groups** tab and then the **Users** subtab.
5. In the table under the Users section, click the name of the user you created in Section 9.2.3.13.6, "Creating Users for the Embedded LDAP," to display the settings section.
6. Select the **Groups** tab.
7. Under Parent Groups, in the Available column, select the role with the read permission (for example, `WikiBlogRO`) that you created in Section 9.2.3.13.3, "Creating Roles (Groups) for the Portal Framework application."
8. Move this role to the Chosen column and click **Save**.
9. Repeat steps 5 to 8 and grant the role with the full access permission to another user you created.

9.2.3.13.9 Migrating Security to a Production Environment

For information about migrating security from a development environment to a production environment, see Section 30.2.5, "Post-deployment Security Configuration Tasks."

9.2.3.13.10 Checking Your Security Group and Roles Configuration

After completing your configuration, follow the steps below to check that the security group and roles have been created correctly, and that a root folder has been created.

To verify that the security group and roles have been created:

1. Log in to the Content Server Console as an administrator.
2. From the Administration menu, select **Admin Applets**.
3. On the Administration Applet page, click **User Admin** to display the User Admin dialog.
4. From the Security menu, select **Permissions by Group**.
5. In the Permission By Group dialog, make sure that the security group is listed in the Groups list. The name of the security group ID should be the same as the Application Name in the Document properties.
6. Select the security group in the groups list.
7. Check that the Roles list contains the two roles: `<applicationName>User` and `<applicationName>AuthenUser` with `R` and `RWD` permissions for the application respectively.

To verify that the root folder has been created:

1. Log in to the Content Server Console as an administrator.
2. From the Browse Content menu, check that the root folder is listed and select it.
3. Verify that the child folder `spacetemplate` is listed
4. Click **Info** to display the Hierarchical Folder Information screen.
5. Verify that the Security Group is correct.

9.2.3.14 Registering the Content Server Repository

Mandatory for Portal Framework applications/Optional, but strongly recommended for WebCenter Portal

For Portal Framework applications, before you can use the configured Content Server, you must configure the connection between the application and Content Server. For WebCenter Portal, although the connection should be configured for you when the application first starts up, you should at least test the connection and check that the expected data has been properly seeded.

This section includes the following subsections:

- Section 9.2.3.14.1, "Configuring a Content Server Connection for Portal Framework applications"
- Section 9.2.3.14.2, "Configuring a Content Server Connection for WebCenter Portal"
- Section 9.2.3.14.3, "Checking the WebCenter Portal Data Seeded in Content Server"

9.2.3.14.1 Configuring a Content Server Connection for Portal Framework applications

After installing and configuring Content Server, continue by configuring the connection between the Portal Framework application and Content Server. For more information about configuring the connection, see Section 9.6.2, "Registering Content Repositories Using Fusion Middleware Control" or Section 9.6.3, "Registering Content Repositories Using WLST."

9.2.3.14.2 Configuring a Content Server Connection for WebCenter Portal

A default connection between WebCenter Portal and Content Server is automatically configured when the application first starts up, however, you should test the connection and check that it has been appropriately configured for your environment. For high availability environments, or for single sign-on environments, you may have to modify the WebCenter Portal host and port settings.

After installing and configuring Content Server, and restarting WebCenter Portal, check the connection between WebCenter Portal and Content Server is properly configured as described in Section 9.11.1, "Testing Content Server Connections." If your connection was not properly configured, then configure it as shown in Section 9.10, "Setting Connection Properties for WebCenter Portal's Default Content Repository."

Some WebCenter Portal components, such as the documents tools, rely on the data seeded in Content Server when WebCenter Portal first starts up. Before configuring other components with WebCenter Portal, check that the expected data has been properly seeded as described in Section 9.2.3.14.3, "Checking the WebCenter Portal Data Seeded in Content Server."

9.2.3.14.3 Checking the WebCenter Portal Data Seeded in Content Server

When WebCenter Portal first starts up, a set of default data is seeded in Content Server. The data seeded in Content Server for a WebCenter Portal instance is based on several properties that are set on the active (or default) content repository connection. For example:

```
Root folder = /WebCenter1
Application Name= WC1
```

If the data is not correct, or has only been partially seeded, check the WebCenter Portal log and your Content Server configuration, make the necessary corrections to these properties, and then restart the WebCenter Portal instance to reseed the data. For

information about setting the default content repository, and setting additional properties required for WebCenter Portal's content repository, see Section 9.10, "Setting Connection Properties for WebCenter Portal's Default Content Repository."

Table 9–3 illustrates the WebCenter Portal data that is seeded (**Seeded Data**), the naming for the data seeded (**Naming**) and how to check that the data is created in Content Server (**Verify**).

Table 9–3 Data Seeded in WebCenter Portal

Seeded Data	Naming	Verify
Security Group	One security group is seeded: *ApplicationName* For example: WC1	In Content Server, go to **Administration > Admin Applets > User Admin > Security> Permission by Group**
Roles	Two roles are seeded: ▪ *ApplicationName* User (with R permission on the security group) ▪ *ApplicationName* AuthenUser (with RWD permission on the security group) For example: WC1User and WC1AuthenUser	In Content Server, go to **Administration > Admin Applets > User Admin > Security > Permission by Role**
Root Folder name	*RootFolder* (with Security Group =<ApplicationName>) For example: /WebCenter1	Browse content (folder will be listed as a top-level folder)
Default Attributes - Public users	All public users have: ▪ Read on the account prefix PUBLIC ▪ Read on the account prefix WCILS ▪ The *ApplicationName*User role	Query the ExtendedConfigProperties table, or after logging into Content Server, click on the username to view the user's profile page listing their roles and accounts, including the account PUBLIC and WCILS and the role <ApplicationName>User
Default Attributes - Authenticated users	All Authenticated users have: ▪ Read permission on the account prefix AUTHEN ▪ Read, Write, Delete,Admin permission on the account prefix WCILS ▪ The *ApplicationName*AuthenUser role	Query the ExtendedConfigProperties table, or after logging into Content Server, click on the username to view the user's profile page listing their roles and accounts, including the account AUTHEN and WCILS and the role *ApplicationName*AuthenUser

Table 9–3 (Cont.) Data Seeded in WebCenter Portal

Seeded Data	Naming	Verify
Workflows	Three workflows are seeded: ■ *ApplicationName*AllApprover ■ *ApplicationName*AllReviewer ■ *ApplicationName*SingleApprover For example, WC1AllApprover, WC1AllReviewer, and WC1SingleApprover	In Content Server, go to **Administration > User Admin > Workflow Admin > Criteria tab**

Table 9–4 illustrates the data that is seeded for the Home portal (**Seeded Data**), the naming for the data seeded (**Naming**) and how to check that the data is created in Content Server (**Verify**). Note that the Home portal data is seeded only once in a Content Server, regardless of how many WebCenter Portal instances are using the same Content Server. Therefore, if you have multiple WebCenter Portal instances using the same Content Server, they will all share the same Home portal data.

Table 9–4 Data Seeded for the Home Portal

Seeded Data	Naming	Verify
Security Group	One security group is seeded: PersonalSpaces	In Content Server, go to **Administration > Admin Applets > User Admin > Security > Permission by Group**
Roles	Two roles are seeded: ■ PersonalSpacesRole (with R permission on the security group PersonalSpaces) ■ PersonalSpacesAuthenRole (with RWD on the security group PersonalSpaces)	In Content Server, go to **Administration > Admin Applets > User Admin > Security > Permission by Role**
Root Folder name	PersonalSpaces (with Security Group=PersonalSpaces)	Browse content (folder will be listed as a top-level folder)
Default Attributes - Public users	All public users have: ■ Read on the Root Folder's account ■ The PersonalSpaces role	Query the ExtendedConfigProperties table, or after logging into Content Server, click on the username to view the user's profile page listing their roles and accounts, including the account PEWebCenter/PU and the role PersonalSpacesRole
Default Attributes - Authenticated users	All Authenticated users have: ■ The PersonalSpacesAuthenRole role	Query the ExtendedConfigProperties table, or after logging into Content Server, click on the username to view the user's profile page listing their roles and accounts, including the role PersonalSpacesAuthenRole

9.3 Configuring a Microsoft SharePoint Repository

If you want to access a Microsoft SharePoint content repository from WebCenter Portal or Portal Framework application, you must install the Oracle WebCenter Adapter for Microsoft SharePoint.

The Oracle WebCenter Adapter for Microsoft SharePoint supports the following features:

- Reading content and metadata from the Microsoft SharePoint repository
- Writing files and folders to the SharePoint document libraries
- Running queries on the Microsoft SharePoint system
- Enabling SharePoint security settings for the accessed content by leveraging native Microsoft SharePoint authentication and authorization

All features are implemented using native Microsoft SharePoint web services as the interface to Microsoft SharePoint content and services.

This section discusses prerequisites for connecting WebCenter Portal or Portal Framework applications to Microsoft SharePoint:

- Section 9.3.1, "Microsoft SharePoint - Installation"
- Section 9.3.2, "Microsoft SharePoint - Configuration"
- Section 9.3.3, "Microsoft SharePoint - Security Considerations"
- Section 9.3.4, "Microsoft SharePoint - Limitations in WebCenter Portal"
- Section 9.3.5, "Managing Microsoft SharePoint Connections Using WLST"

Note: To enable Microsoft SharePoint connections in WebCenter Portal, read the whitepaper "*Integrating the SharePoint 2007 Adapter with WebCenter Spaces*" available from Oracle Technology Network at `http://www.oracle.com/technetwork/middleware/webcenter/owcs-ps3-sharepoint-wcs-wp-335282.pdf`.

9.3.1 Microsoft SharePoint - Installation

This section includes the following:

- Section 9.3.1.1, "About Microsoft SharePoint Server Installation"
- Section 9.3.1.2, "Installing Oracle WebCenter Adapter for Microsoft SharePoint"
- Section 9.3.1.3, "Installing WLST Command Scripts for Managing Microsoft SharePoint Connections"

9.3.1.1 About Microsoft SharePoint Server Installation

Oracle WebCenter Portal supports the following Microsoft SharePoint versions:

- Microsoft Office SharePoint Server (MOSS) 2007 SP2
- Microsoft Windows SharePoint Services (WSS) version 3 SP2

Note: A Microsoft SharePoint site configured for `anonymous` access is not supported by the adapter.

Refer to the appropriate Microsoft SharePoint documentation for installation information.

Oracle WebCenter Portal supports the following Microsoft SharePoint 2007 Document Library version settings:

- Require Check Out: No
- Content Approval: No
- Document Version History: No versioning

If any other version settings are configured, Oracle WebCenter adapter for Microsoft SharePoint does not function correctly. For example, if `Require CheckOut` is set to `yes`, upload operations fail. Similarly, if document version history or content approval is enabled, new versions or documents have restricted visibility.

9.3.1.2 Installing Oracle WebCenter Adapter for Microsoft SharePoint

The files for Oracle WebCenter Adapter for Microsoft SharePoint are located in the Oracle WebCenter Companion DVD in the `ofm_wc_generic_jcr_sharepoint_adapter_11.1.1.4.0.zip` file. When you extract this ZIP file to a temporary location, you will find the adapter files in the *`TEMP_LOCATION`*`/WebCenter/services/content/adapters` directory.

Before You Begin:

The adapter for Microsoft SharePoint must be installed in the same managed server as WebCenter Portal or your Portal Framework application. If you have not done so already, create a managed server suitable for your WebCenter Portal or Portal Framework application deployment.

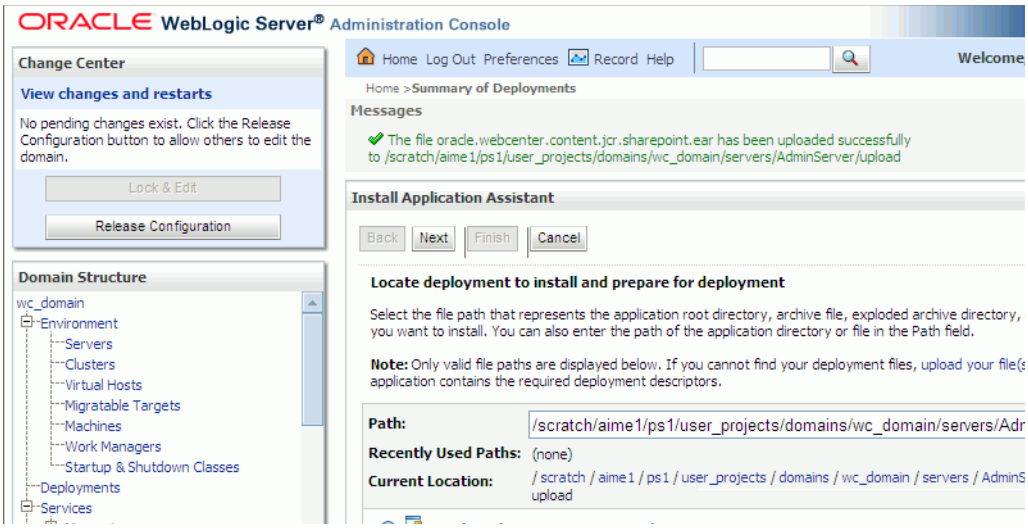
To install WebCenter adapter for Microsoft SharePoint for a WebCenter Portal application:

1. Log in to the WLS Administration Console.

   For information on logging into the WLS Administration Console, see Section 1.13.2, "Oracle WebLogic Server Administration Console."

2. Navigate to the WLS Administration Console's Home page.
3. From the **Domain Structure** pane, click **Deployments**.
4. In the **Summary of Deployments** section, under **Control**, click **Install**.
5. In **Install Application Assistant**, in **Note**, click the **upload your file(s)** link in the body of the text.
6. Click **Browse** next to **Deployment Archive**, select the `oracle.webcenter.content.jcr.sharepoint.ear` file from the *`TEMP_LOCATION`*`/WebCenter/services/content/adapters` directory. This is the temporary directory in which you extracted the contents of the `ofm_wc_generic_jcr_sharepoint_adapter_11.1.1.4.0.zip` file from the **Oracle WebCenter Companion DVD**. Click **Next**.
7. After you see the message that the EAR file has been uploaded successfully, as shown in Figure 9–7, click **Next**.

Figure 9–7 Install Application Assistant



8. Select **Install this deployment as a library**, if not already selected, and click **Next**.
9. In **Select deployment targets**, select the managed server on which WebCenter Portal or your Portal Framework application is deployed.

   For Portal Framework applications, this must be a custom managed server (based on the Custom Portal template), not one of WebCenter Portal's out-of-the-box managed servers. For details, see the "Extending a Domain to Create Custom Managed Servers" section in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*.
10. Click **Next**.
11. In **Optional Settings**, accept the defaults and click **Finish**.

9.3.1.3 Installing WLST Command Scripts for Managing Microsoft SharePoint Connections

1. Extract the files `DocLibSharePointWLST.py` and `DocLibGenericWLST.py` from the **ofm_wc_generic_jcr_sharepoint_adapter_11.1.1.4.0.zip** file located in the Oracle WebCenter Companion DVD. These files are in the `/WebCenter/services/content/adapters` directory.
2. Copy the extracted `DocLibSharePointWLST.py` and `DocLibGenericWLST.py` files and paste them in the *`ORACLE_COMMON_HOME`*`/common/wlst` directory.
3. To run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

For information about managing connections using WLST, see Section 9.3.5, "Managing Microsoft SharePoint Connections Using WLST."

9.3.2 Microsoft SharePoint - Configuration

You must perform the following tasks to enable Microsoft SharePoint connections in WebCenter Portal or Portal Framework applications:

1. Install Oracle WebCenter adapter for Microsoft SharePoint in the same managed server whereWebCenter Portal or your Portal Framework application is (or will be) deployed.

2. For Portal Framework applications:
    1. In JDeveloper, configure a connection to your Microsoft SharePoint repository. This must be an application connection created in Application Resources in the Application Navigator.
    2. (Optional) In JDeveloper, include a Documents task flow that uses the Microsoft SharePoint repository connection.
    3. Deploy your Portal Framework application.

       After deployment, you can access the Microsoft SharePoint repository that you configured in JDeveloper in your Portal Framework application.
3. To reconfigure Microsoft SharePoint connection details for a Portal Framework application postdeployment or to configure a connection for WebCenter Portal:
    a. Install WLST command scripts for managing Microsoft SharePoint connections postdeployment. For details, see Section 9.3.1.3, "Installing WLST Command Scripts for Managing Microsoft SharePoint Connections".
    b. Create a new Microsoft SharePoint repository connection (`createJCRSharePointConnection`) or modify existing connection details (`setJCRSharePointConnection`) as required.

       See also, Section 9.3.5, "Managing Microsoft SharePoint Connections Using WLST".

Note: To enable Microsoft SharePoint connections in WebCenter Portal, read the whitepaper "*Integrating the SharePoint 2007 Adapter with WebCenter Spaces*" available from Oracle Technology Network at `http://www.oracle.com/technetwork/middleware/webcenter/owcs-ps3-sharepoint-wcs-wp-335282.pdf`.

9.3.3 Microsoft SharePoint - Security Considerations

Authentication through identity propagation is not supported on Microsoft SharePoint connections. However, you can use an external application to authenticate users against the Microsoft SharePoint repository. Use the WLST argument `extAppId` to specify the external application to use. For details, see Section 9.3.5.1, "createJCRSharePointConnection." Note that if the `extAppId` refers to an external application connection for which neither public nor shared credentials are defined, then Documents task flows will prompt for credentials. This allows per-user mapping of credentials as an alternative to identity propagation.

9.3.4 Microsoft SharePoint - Limitations in WebCenter Portal

WebCenter Portal does not support Microsoft SharePoint as the default content repository for portals, and therefore, you must use Oracle WebCenter Content instead.

9.3.5 Managing Microsoft SharePoint Connections Using WLST

Use the commands listed in Table 9–5 to manage connections to SharePoint content repositories, postdeployment.

Configuration changes made using these WLST commands are only effective after you restart the Managed Server on which WebCenter Portal or your Portal Framework application is deployed. For details, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

Table 9–5 SharePoint Content Repository WLST Commands

Use this command...	To...	Use with WLST...
createJCRSharePointConnection	Create a Microsoft SharePoint 2007 repository connection.	Online
setJCRSharePointConnection	Edit a Microsoft SharePoint 2007 repository connection.	Online
listJCRSharePointConnections	List all Microsoft SharePoint 2007 connections that are configured for WebCenter Portal or aPortal Framework application.	Online

For information about how to install WLST scripts for Microsoft SharePoint, see Section 9.3.1.3, "Installing WLST Command Scripts for Managing Microsoft SharePoint Connections."

9.3.5.1 createJCRSharePointConnection

The `createJCRSharePointConnection` WLST command creates a connection to a Microsoft SharePoint 2007 repository. For syntax and other information about this WLST command, see "createJCRSharePointConnection" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

> **Note:** For Portal Framework applications, the `createJCRSharePointConnection` command works only if the application was developed to support Microsoft SharePoint connections in the first place. If the original Portal Framework application deployment does not include a Microsoft SharePoint connection, then the application will not contain the code necessary to support any new Microsoft SharePoint connections that you may want to create using this command. See also, Section 9.3.2, "Microsoft SharePoint - Configuration."

9.3.5.2 setJCRSharePointConnection

This WLST command edits an existing Microsoft SharePoint 2007 repository connection. For syntax and other information about this WLST command, see the "setJCRSharePointConnection" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

9.3.5.3 listJCRSharePointConnections

This WLST command lists all of the SharePoint connections that are configured for a named application. For syntax and other information about this WLST command, see the "listJCRSharePointConnections" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

9.4 Configuring an Oracle Portal Repository

This section discusses the prerequisites for an Oracle Portal content repository in the following subsections:

- Section 9.4.1, "Oracle Portal - Installation"
- Section 9.4.2, "Oracle Portal - Configuration"

9.4.1 Oracle Portal - Installation

For information on installing Oracle Portal, see *Oracle Fusion Middleware Installation Guide for Oracle Portal, Forms, Reports and Discoverer*.

9.4.2 Oracle Portal - Configuration

Oracle Portal must be up-to-date with all the latest patches. For additional information about patches, see the product release notes. See also *Oracle Fusion Middleware Administrator's Guide for Oracle Portal*.

9.4.3 Oracle Portal - Security Considerations

None.

9.4.4 Oracle Portal - Limitations in Oracle WebCenter Portal

Oracle Portal integration with Oracle WebCenter Portal is read-only. It is not possible to create content using WebCenter Portal or your Portal Framework application.

You can expose Oracle Portal pages in WebCenter Portal or Portal Framework applications through the Federated Portal Adapter by publishing them as portlets in Oracle Portal. The following are not returned by the Federated Portal Adapter, and therefore not visible in Oracle WebCenter Portal:

- Seeded page groups:
  - Oracle Portal repository
  - Oracle Portal design-time pages
- Pages of the following types:
  - Mobile
  - URL
  - Navigation pages
- Items of the following types:
  - Navigation items
  - PLSQL items
  - Portlet
  - Portlet instance
  - URL items
  - Mobile items
  - Page links
  - Item links
- Items defined as:
  - Expired
  - Hidden

9.5 Configuring a File System Repository

This section discusses the prerequisites for connecting to a file system content repository in the following subsections:

- Section 9.5.1, "File System - Security Considerations"
- Section 9.5.2, "File System - Limitations"

Caution: Only use file system connections during the *development* of Portal Framework applications. File system connections *must not* be used in production or enterprise application deployments. This feature is provided for development purposes only.

WebCenter Portal does not support file system connections.

9.5.1 File System - Security Considerations

All operations are executed as the system user under which the JVM is running and therefore inherit its permissions.

9.5.2 File System - Limitations

File system connections must not be used in production or enterprise application deployments, and search capabilities are limited and slow due to the absence of an index. This feature is provided for development purposes only.

9.6 Registering Content Repositories for WebCenter Portal or Portal Framework Applications

This section contains the following subsections:

- Section 9.6.1, "About Registering Content Repositories for WebCenter Portal"

  Section 9.6.2, "Registering Content Repositories Using Fusion Middleware Control"
- Section 9.6.3, "Registering Content Repositories Using WLST"

9.6.1 About Registering Content Repositories for WebCenter Portal

Consider the following when registering Content Server repositories for WebCenter Portal:

- At start up, WebCenter Portal creates seed data (if it does not already exist) in the primary/active/default repository for WebCenter Portal.
- For WebCenter Portal, a Content Server repository connection must always be provided as a primary connection, even if another repository such as Microsoft SharePoint is made available.
- A user name with administrative rights for the Content Server instance is required (`Content Administrator`). This user will be used to create and maintain folders for portal content, security groups and roles, and manage content access rights. The default content administrator is `sysadmin`.

  Administrative privileges are required for this connection so that operations can be performed on behalf of WebCenter Portal users.

- `Root Folder` and `Application Name` values:
  - For the active connection in WebCenter Portal, the `Root Folder` and `Application Name` values are used to create the seed data in the WebCenter Portal repository to enable storage of portal-related data.

    WARNING: You should never change the `Root Folder` or `Application Name` values separately; you should always change both. That is, if you change the `Root Folder` value after configuring and running WebCenter Portal, then you must also change the `Application Name` value, and vice versa. That is, you must change both values (`Root Folder` and `Application Name`) to unique values if WebCenter Portal already contains the seed data.

    When you change these values, the existing seed data is not renamed in the Content Server repository. Instead, new seed data is created using the new values, when you start the application. Once the application is started, new WebCenter Portal data is created under the new `Root Folder` and existing data under the old `Root Folder` is no longer available. This means that the documents tools will now be disabled in WebCenter Portal where the documents tools were previously enabled, prior to changing the `Root Folder`.

    Note: Although the `Root Folder` and `Application Name` values change, the old root content repository folder still appears in search results, like any other root folder in Content Server.

  - The `Root Folder` value is used as the name for the root folder within the content repository under which all WebCenter Portal content is stored. For the `Root Folder` value, you must specify a content repository folder that does not yet exist. Use the format: `/foldername`. For example: `/MyWebCenterSpaces`. The `Root Folder` cannot be `/`, the root itself, and it must be unique across different portals. The folder specified is created for you when the WebCenter Portal starts up. Invalid entries include: `/`, `/foldername/`, `/foldername/subfolder`.
  - The `Application Name`, identifies a WebCenter Portal instance within this content repository and must have a unique value (for example: `MyWCPApp`). The name must be 14 characters or less, begin with an alphabetical character, followed by any combination of alphanumeric characters or the underscore character. The name specified here is also used to name document-related workflows, as follows: `<applicationName><WorkflowName>` and `<applicationName><WorkflowStepName>`. When naming workflows, only the first 14 characters of the `Application Name` are used.

    The `Application Name` value is used for the following:

    * To separate data when multiple WebCenter Portal instances share the same content repository and should be unique across applications.
    * As the prefix to the seeded workflow and workflow steps.
    * As the name of the security group in which all data created in that WebCenter Portal instance is stored.

* As the prefix for the role (the name format is `applicationNameUser` and `applicationNameAuthenUser`).
* To stripe users permissions on accounts for the particular WebCenter Portal instance.
* To stripe default attributes for the particular WebCenter Portal instance.

For information about security groups and roles, see the "Advanced Administration: Security" part in *Oracle Fusion Middleware Administering Oracle WebCenter Content*. For information about folders, see the "Organizing Content" chapter in *Oracle Fusion Middleware Managing Oracle WebCenter Content*.

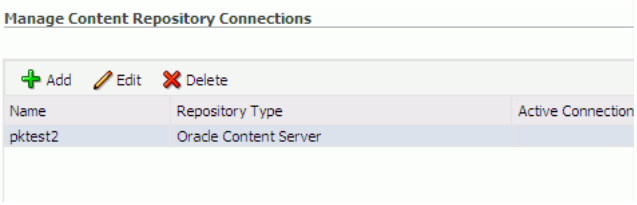
9.6.2 Registering Content Repositories Using Fusion Middleware Control

Follow the steps below to register a Content Server, Oracle Portal, or file system content repository using Fusion Middleware Control. Note that to register a SharePoint repository you must use WLST as described in Section 9.6.3, "Registering Content Repositories Using WLST." For information on how to register a Content Server repository using WLST, see Section 9.10.2, "Setting Connection Properties for WebCenter Portal's Default Content Repository Using WLST."

To register a Content Server, Oracle Portal, or file system content repository:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal or Portal Framework application:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For WebCenter Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.
3. From the list of services on the WebCenter Service Configuration page, select **Content Repository**.
4. To connect to a new content repository, click **Add** (Figure 9–8).

Figure 9–8 Configuring Content Repository Connections



5. Enter a unique name for this connection, specify the content repository type, and indicate whether this connection is the active (or default) connection for the application. See Table 9–6.

Table 9–6 Manage Content Repository Connections

Field	Description
Connection Name	Enter a unique name for this content repository connection. The name must be unique (across all connection types) within WebCenter Portal or Portal Framework application.
Repository Type	Choose the type of repository you want to connect to. Select one of the following: ■ **Content Server** - an Oracle Universal Content Management repository. See Section 9.2, "Configuring a Content Server Repository". ■ **Oracle Portal** - an Oracle Portal content repository. See Section 9.4, "Configuring an Oracle Portal Repository." ■ **File System** - a computer file system. See Section 9.5, "Configuring a File System Repository." **Caution:** File system connections *must not* be used in production or enterprise application deployments. This feature is provided for development purposes only. (WebCenter Portal) If you are setting up the back-end content repository for WebCenter Portal, that is, the repository used to store portal-related documents, you must select **Content Server**.
Active Connection	Select to make this the *default* or *primary* content repository for WebCenter Portal or your Portal Framework application. You can connect WebCenter Portal or your Portal Framework application to multiple content repositories; all connections are used. One connection must be designated the *default* (or active) connection. Do one of the following: ■ For WebCenter Portal: Select to make this the *active connection* for the back-end repository that WebCenter Portal uses to store portal-related documents. The active connection must be to a Content Server repository. If this is the *default content repository* for WebCenter Portal, some additional configuration is required -- see Section 9.10.1, "Setting Connection Properties for WebCenter Portal's Default Content Repository Using Fusion Middleware Control." ■ For Portal Framework applications: Select to make this the *active connection*; that is, the default connection for Content Presenter, Document Manager, Document List Viewer, and Recent Documents task flows. When no specific connection details are provided for these task flows, this default (also called *primary*, *active*) connection is used. Deselecting this option does not disable the content repository connection. If a content repository is no longer required, you must delete the connection.

6. (For the active connection in WebCenter Portal only.) Enter additional details for the WebCenter Portal repository. For information, see Section 9.6.1, "About Registering Content Repositories for WebCenter Portal."
7. Enter connection details for the content repository. For detailed parameter information, see:
   - Table 9–7, " Content Server Connection Parameters"

- Table 9–8, " Connection - Content Server - Cache Details"
- Table 9–9, " Oracle Portal Connection Parameters"
- Table 9–10, " File System Connection Parameters"

Table 9–7 Content Server Connection Parameters

Field	Description
RIDC Socket Type	Specify whether Content Server connects on the content server listener port or the Web server filter, and whether the listener port is SSL enabled. Choose from: ▪ **Socket** - Uses an `intradoc` socket connection to connect to the Content Server. The client IP address must be added to the list of authorized addresses in Content Server. In this case, the client is the machine on which Oracle WebCenter Portal is running. ▪ **Socket SSL** - Uses an `intradoc` socket connection to connect to Content Server that is secured using the SSL protocol. The client's certificates must be imported in the server's trust store for the connection to be allowed. This is the most secure option, and the recommended option whenever identity propagation is required (for example, in WebCenter Portal). ▪ **Web** - Uses an HTTP(S) connection to connect to Content Server. ▪ **JAX-WS** - Uses an HTTP(S) connection to connect to Content Server. For WebCenter Portal, the **Web** option is not suitable for the active connection, that is, the back-end Content Server repository that is being used to store portal-related documents because it does not allow identity propagation.
Server Host	Enter the host name of the machine where Content Server is running. For example: `mycontentserver.mycompany.com` Server Host is required when the **RIDC Socket Type** is set to **Socket** or **Socket SSL**.
Server Port	Enter the port on which the Content Server listens: ▪ **Socket** - Port specified for the `incoming` provider in the server. ▪ **Socket SSL** - Port specified for the `sslincoming` provider in the server. This property corresponds to the IntradocServerPort setting in the Content Server configuration file, which defaults to port `4444`. You can find the current value by logging onto the Content Server and navigating to **Administration > Admin Server > General Configuration >Additional Configuration Variables > IntradocServerPort**. Server Port is required when the RIDC Socket Type is set to **Socket** or **Socket SSL**.
Web URL	Enter the Web server URL for the Content Server. Use the format: `http://`*`hostname`*`:`*`portnumber`*`/`*`web_root`*`/`*`plugin_root`* For example: `http://mycontentserver/cms/idcplg` Web URL is applicable when the RIDC Socket Type is set to **Web**.

Table 9–7 (Cont.) Content Server Connection Parameters

Field	Description
Web Service URL	Enter the Web service URL required to connect to Content Server when using the JAX-WS protocol. Use the format: *`http://hostname:port/web_root`* For example: `http://myhost.com:9044/idcnativews` Web Service URL is applicable when RIDC Socket Type is set to **JAX-WS**.
Connection Timeout (ms)	Specify the length of time (in milliseconds) allowed to log in to WebCenter Content Server before issuing a connection timeout message, and set the socket timeout for the underlying RIDC connection for all service requests. **Note:** The RIDC socket timeout only applies to **Socket**, **Socket SSL**, and **Web** connection types. If the **Connection Timeout** property is not set, the following values are used: ▪ **Login timeout** - the default concurrency timeout is configure for the oracle.webcenter.content resource is used (30s or 30000ms). Refer to "Configuring Concurrency Management" in *Oracle Fusion Middleware Performance and Tuning Guide* for more information. ▪ **RIDC socket timeout** - the default RIDC socket timeout (60s or 60000ms) is used for all service requests for connection types Socket, Socket SSL, and Web. If the Connection Timeout property is set and the connection type is **Socket**, **Socket SSL**, or **Web**, Oracle recommends that you do not specify a value less than 60000ms, as this would reduce the RIDC socket timeout and increase the likelihood that long running requests time out. For example, timeouts may occur during long running searches, long file uploads, or long copy operations.
Authentication Method	Choose from: ▪ **Identity Propagation** - Content Server and WebCenter Portal (or Portal Framework application) use the same identity store to authenticate users. (WebCenter Portal) Identity propagation is required on the active connection for WebCenter Portal (that is, for the content repository being used to store portal-related documents). ▪ **External Application** - An external application authenticates users against the Content Server. Select this option if you want to use public, shared, or mapped credentials. See also, "Setting Security for the Documents Tool" in the *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*. If an external application is used for authentication, use the **Associated External Application** drop down list to identify the application. If the application you want is not listed, select **Create New** to define the external application now.

Table 9–7 (Cont.) Content Server Connection Parameters

Field	Description
Web Server Context Root	Enter the Web server context root for Content Server. Use the format `/<context_root>`. For example, `/cs`. When specified, several Content Server features based on iFrame are available in WebCenter Portal or your Portal Framework application. This includes: ▪ **Associating a content profile with files when uploading new or updated files to Content Server.** For more information, see the "Uploading Files" and "Uploading a New Version of an Existing File" sections in *Oracle Fusion Middleware Using Oracle WebCenter Portal*. ▪ **Using the document review functionality available in Oracle AutoVue.** For more information, see the "Collaborating on Documents Using Oracle AutoVue" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*. ▪ **Editing advanced document properties.** For more information, see the "Viewing Files in Workflow" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*. ▪ **Viewing folder and file workflow details.** For more information, see the "Viewing Files in Workflow" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*. ▪ **Previewing files in a slide viewer.** For more information, see the "Opening a File" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*. ▪ **Site Studio integration** Without OHS (and WebContextRoot configuration), it is still possible to create or edit Site Studio content from within Content Presenter, but the create and edit actions launch new browser windows (or tabs) rather than opening within the Content Presenter task flow. For more information, see the "Creating or Editing Site Studio Content in the Content Presenter Configuration Dialog" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*. The Web Server Context Root property is only applicable when the Authentication Method is set to **Identity Propagation**. **Note:** Specifying the Web Server Context Root is an indicator that WebCenter Portal or your Portal Framework application is front-ended by OHS. If you specify the Web Server Context Root and do not connect through OHS, a 404 error occurs while you attempt to edit the advanced metadata in the Document Viewer, upload using a profile, or click Details for a content item in a workflow in a portal. For information about setting up OHS to front-end WebCenter Portal or Portal Framework applications, see Appendix B, "Oracle HTTP Server Configuration for WebCenter Portal." If WebCenter Portal or your Portal Framework application is connected to multiple Content Server servers, Oracle recommends that each Content Server server has a unique Web Server Context Root so that OHS re-direction works correctly.

Table 9–7 (Cont.) Content Server Connection Parameters

Field	Description
Associated External Application	Select the external application used to authenticate users against Content Server. Associated External Application is applicable when RIDC Socket Type is set to **Web** and also when the RIDC Socket Type is **Socket** or **Socket SSL** (with Authentication Method set to **External Application**).
Client Security Policy	Enter the client security policy to be used when the **RIDC Socket Type** is **JAX-WS**. For example: `oracle/wss11_saml_token_with_message_protection_service_policy` The JAX-WS client security policy can be any valid OWSM policy, but must match the security policy configured for the Content Server's native web service (`IdcWebLoginService`). For more information, see the "WebCenter Content Web Services" section in *Oracle Fusion Middleware Developing with Oracle WebCenter Content*. Leave this field blank if your environment supports Global Policy Attachments (GPA).
Administrator User Name	Enter a user name with administrative rights for this Content Server instance. This user will be used to fetch content type information based on profiles and track document changes for cache invalidation purpose. Defaults to `sysadmin`.
Administrator Password	Enter the password for the Content Server administrator.
Key Store Location	Specify the location of key store that contains the private key used to sign the security assertions. The key store location must be an absolute path. For example: `D:\keys\keystore.xyz` Key Store Location is required when the RIDC Socket Type is set to **Socket SSL**.
Key Store Password	Enter the password required to access the keystore. For example: `T0PS3CR3T` Key Store Password is required when the RIDC Socket Type is set to **Socket SSL**.
Private Key Alias	Enter the client private key alias in the keystore. The key is used to sign messages to the server. The public key corresponding to this private key must be imported in the server keystore. Ensure that the alias does not contain special characters or white space. For example: `enigma` Private Key Alias is required when the RIDC Socket Type is set to **Socket SSL**.
Private Key Password	Enter the password to be used with the private key alias in the key store. For example: `c0d3bR3ak3R` Private Key Password is required when the RIDC Socket Type is set to **Socket SSL**.

Table 9–8 Connection - Content Server - Cache Details

Element	Description
Cache Invalidation Interval (minutes)	Specify the time between checks for external Content Server content changes (in minutes). WebCenter Portal automatically clears items that have changed from the cache. The *minimum* interval is 2 minutes. By default, cache invalidation is disabled so no periodic check made for content changes (shown as `0`).
Maximum Cached Document Size (bytes)	Enter a maximum cacheable size (in bytes) for Content Server binary documents. Documents larger than this size are not cached by WebCenter Portal. The default is 102400 bytes (100K). Tune this value based on your machine's memory configuration and the types of binary documents that you expect to cache. **Note:** Most documents stored in Content Server are considered binary content, that is, images, plain text, Word documents, and so on. The only exception is Site Studio content which is stored in CDF data files and cached separately in a Virtual Content Repository (VCR) cache (or node cache).

Table 9–9 Oracle Portal Connection Parameters

Field	Description
Data Source Name	Enter the JNDI DataSource location used to connect to Oracle Portal. For example: `jdbc/MyPortalDS` The data source must be on the server where WebCenter Portal or your Portal Framework application is deployed.
Connection Timeout (ms)	Specify the length of time (in milliseconds) allowed to log in to Oracle Portal before issuing a connection timeout message. If no timeout is set, the default concurrency timeout for the oracle.webcenter.content resource is used (30s or 30000ms). Refer to "Configuring Concurrency Management" in *Oracle Fusion Middleware Performance and Tuning Guide* for more information.
Authentication Method	Specify how to authenticate users against Oracle Portal. Choose from: ■ **Identity Propagation** - Select this option when WebCenter Portal (or your Portal Framework application) and Oracle Portal both use the same user identity store. ■ **External Application** - Use an external application to authenticate users against Oracle Portal. Select this option if you want to use public, shared, or mapped credentials. If an external application is used for authentication, use the **Associated External Application** dropdown list to identify the application.
Associated External Application	Associate Oracle Portal with an external application. External application credential information is used to authenticate Oracle Portal users. You can select an existing external application from the dropdown list, or click **Create New** to configure a new external application now.

Table 9–10 File System Connection Parameters

Field	Description
Base Path	Enter the full path to a folder on a local file system in which your content is placed. For example: `C:\MyContent` **Caution:** File system content *must not* be used in production or enterprise application deployments. This feature is provided for development purposes only.

8. Click **OK** to save this connection.
9. Click **Test** to verify if the connection you created works. For a successful connection, the Test Status message displays the advice that to start using the new (active) connection, you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed.

The registered connections are now available to Documents and Content Presenter task flows, which you can add to pages in WebCenter Portal or your Portal Framework application. See also, "Working with Document Task Flows and Document Components" in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

9.6.3 Registering Content Repositories Using WLST

Use the following WLST commands to register new content repository connections for Portal Framework applications:

- **Content Server** - `createJCRContentServerConnection`
- **File System** - `createJCRFileSystemConnection`
- **Oracle Portal** - `createJCRPortalConnection`
- **Microsoft SharePoint** - createJCRSharePointConnection

For command syntax and examples, see the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

To configure a particular connection as the default connection, set `isPrimary='1'`. See Section 9.7, "Changing the Active (or Default) Content Repository Connection."

> **Note:** You must set additional connection properties for WebCenter Portal's *default* Content Server repository, see Section 9.10.2, "Setting Connection Properties for WebCenter Portal's Default Content Repository Using WLST."

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

> **Note:** To start using the new (active) connection you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed. See "Starting and Stopping Managed Servers Using WLST" in the *Oracle Fusion Middleware Administrator's Guide*. Note that if you are using the documents tools, Content Server should be started first to allow for initial provisioning to take place.

9.7 Changing the Active (or Default) Content Repository Connection

WebCenter Portal and Portal Framework applications support multiple content repository connections but only one content repository connection can be designated the active (or default) connection.

In WebCenter Portal, the *active connection* becomes the default back-end repository for portal documents (including the Home portal) and the repository must be Content Server. The *active connection* is also used as the default connection for Documents and Content Presenter task flows.

For Portal Framework applications, the *active connection* becomes the default connection for Content Presenter, Document Manager, Document List Viewer, and Recent Documents, and so on. When no specific connection details are provided for these task flows, the default (active) connection is used.

This section contains the following subsections:

- Section 9.7.1, "Changing the Active (or Default) Content Repository Connection Using Fusion Middleware Control"
- Section 9.7.2, "Changing the Active (or Default) Content Repository Connection Using WLST"

9.7.1 Changing the Active (or Default) Content Repository Connection Using Fusion Middleware Control

To change the active (or default) content repository connection:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal or the Portal Framework application:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings > Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal > Service Configuration**.
3. From the list of services on the WebCenter Portal Services Configuration page, select **Content Repository**.

   The Manage Content Repository Connections table indicates the current active connection (if any).
4. Select the connection you want to become the active (or default) connection, and then click **Edit**.
5. Select the **Active Connection** checkbox.

   Before saving your changes, be sure to provide values for the **Content Administrator**, **Root Folder** and **Application Name** fields.
6. Click **OK** to update the connection.
7. Click **Test** to verify if the connection you activated works. For a successfully activated connection, the Test Status message displays the advice that to start

using the updated connection you must restart the managed server on which the WebCenter Portal or your Portal Framework application is deployed.

9.7.2 Changing the Active (or Default) Content Repository Connection Using WLST

Use the following WLST commands with `isPrimary='1'` to designate an existing content repository connection as the default connection:

- **Content Server** - `setJCRContentServerConnection`
- **File System** - `setJCRFileSystemConnection`
- **Oracle Portal** - `setJCRPortalConnection`
- **Microsoft SharePoint** - setJCRSharePointConnection

  See also, listJCRSharePointConnections

For command syntax and examples, see the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

To subsequently disable a default content repository connection, run the same WLST command with `isPrimary='false'`. Connection details are retained but the connection is no longer named as the primary connection in `adf-config.xml`.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Note: To start using the new (active) connection you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed. See, "Starting and Stopping Managed Servers Using WLST" in the *Oracle Fusion Middleware Administrator's Guide*.

9.8 Modifying Content Repository Connection Details

This section contains the following subsections:

- Section 9.8.1, "Modifying Content Repository Connection Details Using Fusion Middleware Control"
- Section 9.8.2, "Modifying Content Repository Connection Details Using WLST"
- Section 9.8.3, "Modifying Cache Settings for Content Presenter"

9.8.1 Modifying Content Repository Connection Details Using Fusion Middleware Control

To update content repository connection details:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal or the Portal Framework application:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.

- For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.

3. From the list of services on the WebCenter Portal Services Configuration page, select **Content Repository**.
4. Select the connection name, and click **Edit**.
5. Edit connection details, as required. For detailed parameter information, see:
   - Table 9–7, " Content Server Connection Parameters"
   - Table 9–9, " Oracle Portal Connection Parameters"
   - Table 9–10, " File System Connection Parameters"
6. Click **OK** to save your changes.
7. Click **Test** to verify if the updated connection works. For a successfully updated connection, the Test Status message displays the advice that to start using the updated connection, you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed.

9.8.2 Modifying Content Repository Connection Details Using WLST

Use the following WLST commands to edit content repository connections:

- **Oracle WebCenter Content Server** - `setJCRContentServerConnection`
- **File System** - `setJCRFileSystemConnection`
- **Oracle Portal** - `setJCRPortalConnection`

For command syntax and examples, see the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

To configure a particular connection as the active (or default) connection, set `isPrimary='1'`. See Section 9.7, "Changing the Active (or Default) Content Repository Connection."

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Note: To start using the updated (active) connection details, you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed. See "Starting and Stopping Managed Servers Using WLST" in the *Oracle Fusion Middleware Administrator's Guide*.

9.8.3 Modifying Cache Settings for Content Presenter

Out of the box, Content Presenter uses local (in-memory) caches. Oracle recommends, however, that Coherence be used as the caching mechanism for production environments, and Coherence is required for high availability environments. You can enable Coherence for caches by modifying its configuration file, and for Portal Framework applications, also adding it to the application (EAR) classpath or server's system classpath. For more information about configuring Coherence for caches, see the "Configuring Coherence as the Caching Mechanism" section in the *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

9.9 Deleting Content Repository Connections

This section contains the following subsections:

- Section 9.9.1, "Deleting Content Repository Connections Using Fusion Middleware Control"
- Section 9.9.2, "Deleting Content Repository Connections Using WLST"

> **Caution:** Delete a content repository connection only if it is not in use. If a connection is marked as active, it should first be removed from the active list, and then deleted.

9.9.1 Deleting Content Repository Connections Using Fusion Middleware Control

To delete a content repository connection:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal or the WebCenter Portal Framework application:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.
3. From the list of services on the WebCenter Portal Services Configuration page, select **Content Repository**.
4. Select the connection name, and click **Delete**.
5. To effect this change you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed.

9.9.2 Deleting Content Repository Connections Using WLST

Use the WLST command `deleteConnection` to remove a content repository connection. For command syntax and examples, see the "deleteConnection" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

> **Note:** To effect this change you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed. See, "Starting and Stopping Managed Servers Using WLST" in the *Oracle Fusion Middleware Administrator's Guide*.

9.10 Setting Connection Properties for WebCenter Portal's Default Content Repository

You can view, modify, and delete connection properties for the back-end Content Server repository that is being used by WebCenter Portal to store portal documents (including the Home portal documents). Specifically, you can define the root folder under which portal content is stored, the name of the content repository administrator, and a unique application identifier for separating application data on Content Server.

This section contains the following subsections:

- Section 9.10.1, "Setting Connection Properties for WebCenter Portal's Default Content Repository Using Fusion Middleware Control"
- Section 9.10.2, "Setting Connection Properties for WebCenter Portal's Default Content Repository Using WLST"

9.10.1 Setting Connection Properties for WebCenter Portal's Default Content Repository Using Fusion Middleware Control

To set content repository connection properties for WebCenter Portal:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal. See Section 6.2, "Navigating to the Home Page for WebCenter Portal."
2. From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
3. From the list of services on the WebCenter Portal Services Configuration page, select **Content Repository**.
4. Select the connection name, and click **Edit**.
5. (For the active connection in WebCenter Portal only.) Set connection properties for the WebCenter Portal repository. For information, see Section 9.6.1, "About Registering Content Repositories for WebCenter Portal."
6. Click **OK** to save your changes.
7. To start using the updated (active) connection properties, you must restart the managed server on which WebCenter Portal is deployed (`WC_Spaces` by default).

9.10.2 Setting Connection Properties for WebCenter Portal's Default Content Repository Using WLST

The following commands are valid only for the WebCenter Portal application to view, set, and delete properties for the Content Server repository that is being used by WebCenter Portal to store portal documents:

- `listDocumentsSpacesProperties`
- `setDocumentsSpacesProperties`
- `deleteDocumentsSpacesProperties`

For command syntax and detailed examples, see the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

9.11 Testing Content Repository Connections

After setting up content repository connections, you can test them to make sure that you can access the content repository, as described in the following sections:

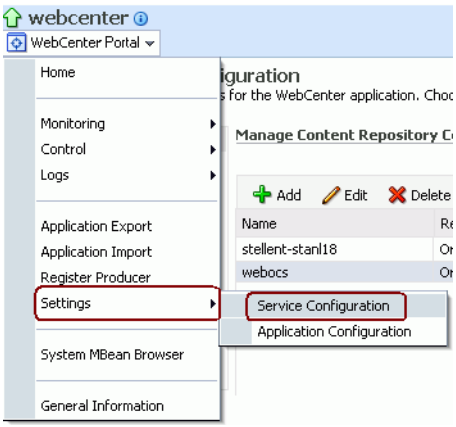
- Section 9.11.1, "Testing Content Server Connections"
- Section 9.11.2, "Testing Oracle Portal Connections"

9.11.1 Testing Content Server Connections

To verify a connection of the socket type `web`, log in to the Web interface of Content Server as `administrator`. You can obtain the URL of a socket type connection through Fusion Middleware Control as follows:

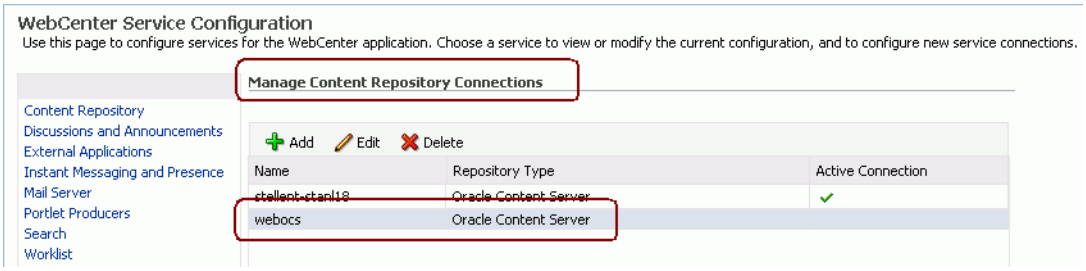
1. In Fusion Middleware Control, from the **WebCenter Portal** menu, select **Settings** and select **Service Configuration** (Figure 9–9).

Figure 9–9 Fusion Middleware Control WebCenter Portal Menu



2. On the **Manage Content Repository Connections** page, select the connection and click **Edit** (Figure 9–10).

Figure 9–10 Manage Content Repository Connections Page

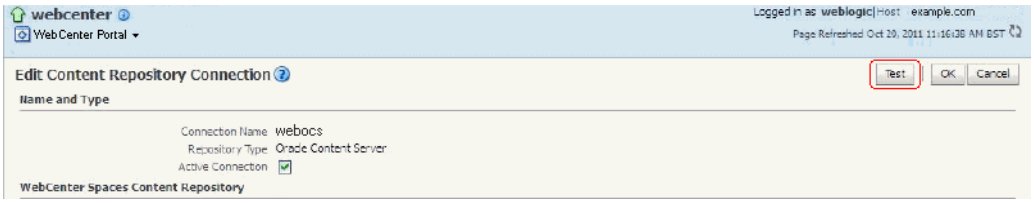


3. On the **Edit Content Repository Connection** page, copy the Web URL (Figure 9–11).

> **Note:** Remove the `/idcplg/` suffix from the URL before using it.

The URL format is: `http://host_name/web_root/`
For example: `http://mycontentserver/cms/`

Figure 9–11 Edit Content Repository Connection Page

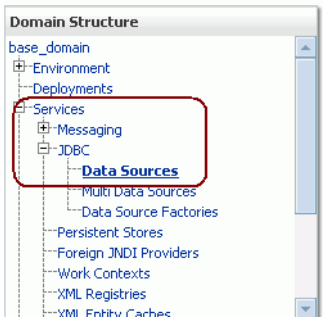


9.11.2 Testing Oracle Portal Connections

To verify the full state of an Oracle Portal connection:

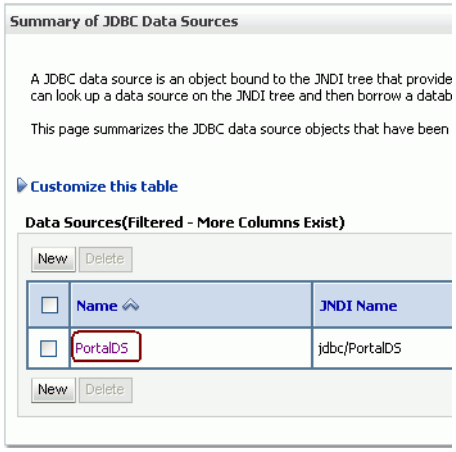
1. In the Oracle WebLogic Administration Console, under **Domain Structure**, expand **Services** > **JDBC**, then double-click **Data Sources** (Figure 9–12).

Figure 9–12 Oracle WebLogic Administration Console



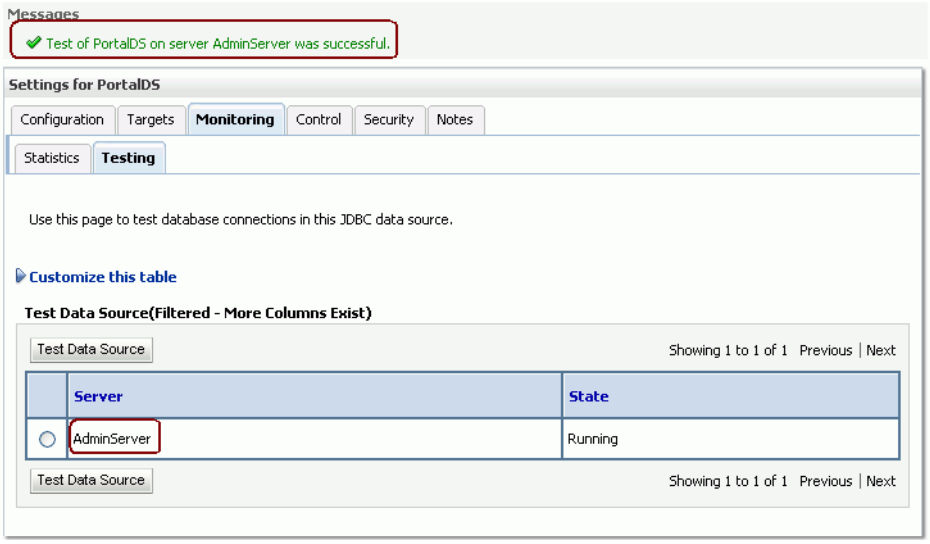
2. On the **Summary of JDBC Data Sources** page, select the data source you intend to test (Figure 9–13).

Figure 9–13 Summary of JDBC Data Sources Page



3. In the **Settings for** *datasource_name* section, select the tabs **Monitoring**, then **Testing**. Select the data source target server, then click **Test Data Source** to test the connection (Figure 9–14).

Figure 9–14 Data Source Settings Section



9.12 Changing the Maximum File Upload Size

By default, the maximum upload size for files is:

- 2 MB for Portal Framework applications. This default is imposed by Apache MyFaces Trinidad, which handles uploading files from a browser to the application server.

  Portal Framework application developers can customize the maximum file upload size at design time by setting the `org.apache.myfaces.trinidad.UPLOAD_MAX_DISK_SPACE` parameter in the `web.xml` file. For more information, see the "Setting Parameters to Upload Files to Content Repositories" section in the *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*. See also Section A.1.2, "web.xml."

- 2 GB for WebCenter Portal.

  System administrators can customize the maximum file upload size by editing the `uploadedFileMaxDiskSpace` parameter in the `webcenter-config.xml` file. For details, see Section A.1.3, "webcenter-config.xml."

9.13 Troubleshooting Issues with Content Repositories

This section includes the following subsections:

- Section 9.13.1, "Documents Service Unavailable In WebCenter Portal"

9.13.1 Documents Service Unavailable In WebCenter Portal

If document tools are not available in WebCenter Portal, that is, the Documents tab is not available in your Home portal or other portals, there may be a connection issue to the backend Content Server, or the Content Server does not contain some required WebCenter Portal data.

To diagnose the problem, follow these steps:

1. Check that the Content Server is up and running. Ensure the server has the **Server Port** (`intradoc`) configured and the **Server IP Filter** allows WebCenter Portal to connect:
   a. Log in to the Content Server.
   b. Click **Administration**.
   c. Click **Configuration for <instance name>**.
   d. Click the **Server Configurations** link under System Configuration.
   e. Ensure that **Server Port** is listed and that **Server IP Filter** allows access from WebCenter Portal.
2. Check the connection between WebCenter Portal and the WebCenter Content Server that is being used as the backend document store:
   a. Login to Fusion Middleware Control, and navigate to Content Repository Connection settings. For details, see Section 9.6.2, "Registering Content Repositories Using Fusion Middleware Control."
   b. Select and edit the required connection.
   c. Ensure the **Active Connection** check box is selected.
   d. Ensure that **Content Administrator**, **Root Folder** and **Application Name** are specified correctly:
      – **Content Administrator** - the Content Administrator *must* have administration rights on the Content Server. This user will be used to create and maintain folders for portal content, security groups and roles, and manage content access rights.
      – **Root Folder and Application Name** - both must be unique and not used by any other WebCenter Portal instance using the same Content Server. If you change these values, ensure that both values are changed and not just one of them.
      – **Application Name** - must be 14 characters or less as it is used as a prefix for items created in Content Server, such as workflows, which have a limit on the length of the item name.
   e. If you make changes, click **OK** to save the connection.
   f. Click **Test** to verify that the connection works.
   g. If you made changes, you must restart `WC_Spaces`, the managed server on which WebCenter Portal is deployed.
   h. Log in to WebCenter Portal to see if the documents tools are available after your connection updates.
3. If the Document service is still not available, check log messages around WebCenter Portal start-up for any errors connecting to the Content Server or saving data on the Content Server.

   For details, see Section 28.2.1, "Viewing and Configuring WebCenter Portal Logs."
4. If the log does not show any useful log information, increase the logging level for the Content Server, and then restart WebCenter Portal to investigate the messages in more detail:
   a. Use Fusion Middleware Control (or edit the `logging.xml` file) to increase logging for `oracle.webcenter.doclib.internal.model` and `oracle.webcenter.doclib.internal.spaces`.

See also, Section 28.2.1, "Viewing and Configuring WebCenter Portal Logs."

b. Restart WebCenter Portal.

c. View the logs again:

If WebCenter Portal data already exists on Content Server, the following message is logged at TRACE level:

```
Content Server already contains the Space container, therefore no need
to seed any data
```

If WebCenter Portal data is not yet seeded, the following message is logged at TRACE level:

```
Creating WebCenter Seeded Data
```

10

Managing Activity Graph

This chapter describes how to configure and manage activity graph for WebCenter Portal and Portal Framework applications.

> **Note:** Always use the activity graph administration, Fusion Middleware Control, or WLST command-line tool to review and configure the activity graph. Oracle does not recommend that you edit files manually (unless specifically instructed to do so) as this can lead to misconfiguration.

This chapter includes the following topics:

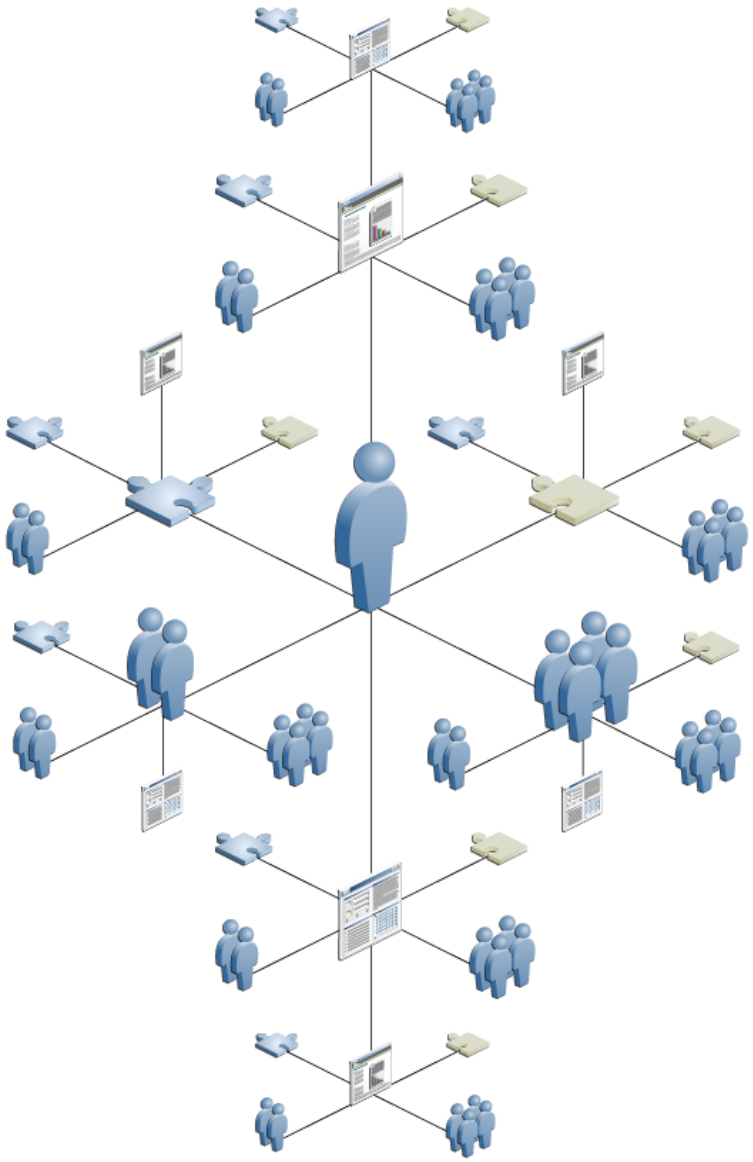
> **Permissions:** To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console and the `Administrator` role in the deployed application:
>
> - **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.
> - **Portal Framework application**: `Administrator` role granted through the Administration Console.
>
> For more information about roles and permissions, see Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

10.1 About Activity Graph

Activity graph provides suggestions of people that a user may be interested in connecting with, based on existing connections and shared interaction with objects in the application. It also directs users to portals or content that may be of interest, based on similar interactions with those portals, items that the user is currently viewing, or the most active items.

The activity graph presents these suggestions based on data gathered and analyzed by the activity graph engines. The activity graph engines provide a central repository for actions that are collected by enterprise applications. Thinking in terms of a mathematical graph, application users and the enterprise content with which they interact are *nodes*, and the actions between users and between users and content are *directed edges* (Figure 10–1).

Figure 10–1 An Activity Graph

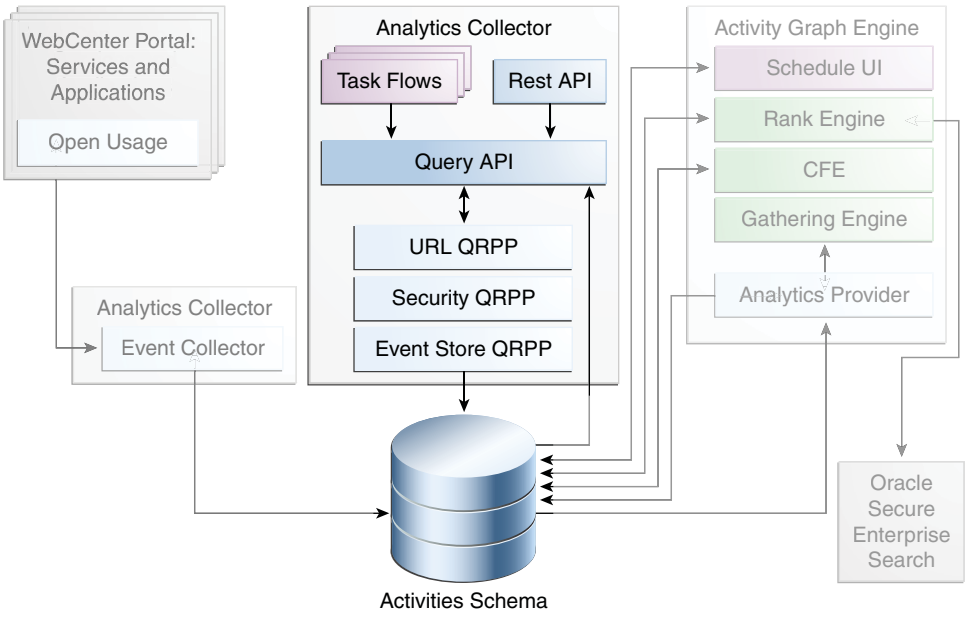


There are three main components used by the activity graph:

- The Oracle Analytics Event Collector collects event data using the OpenUsage API and saves that data in the Activities database.
- The activity graph engines include engines for gathering data, calculating similarity scores, and calculating search rankings.
- The activity graph query API exposes raw action data and processed recommendation data, and mechanisms (QRPPs) through which to filter results from people who do not have access to see them and decorate results with up-to-date metadata.

Figure 10–2 shows how the different components work together. The process is described below.

Figure 10–2 Activity Graph Architecture



When an action occurs in WebCenter Portal, for example, Monty viewing his document, it is picked up by the Analytics Events Collector and placed in an event table in the Activities database.

When the activity data gathering process starts, the Analytics Activity Provider reads actions from the Analytics event tables and uses a registered set of mappings to generate activities. An *activity* is one occurrence of an action and is used to determine *relations*, aggregated occurrences of actions, which are stored in the relation tables. For example, the fact that Monty has viewed this particular document five times is a relation. Information in the relation tables is used to determine recommendations and search ranks.

The activity graph query API is a Java API, used by the activity graph task flows, that queries the relation tables for recommendations using a recipe. A *recipe* is a weighted list of rank or similarity calculations. A *similarity calculation* provides a similarity score (a number between zero and one) that designates how similar two objects are to each other given a specific criterion. The weighting of each calculation determines its significance in deciding the overall recommendation score. Recommendations are ordered by their total recommendation score. WebCenter Portal provides default calculations, used by the activity graph task flows. You can edit these calculations or create custom calculations. For more information, see the "Defining Custom Similarity

Calculations" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

After the initial list of recommendations for a particular object is generated, the results can be filtered into something more appropriate and useful to present to users. This is achieved using Query Result Post-Processors (QRPPs). QRPPs take the current list of recommendation results return a modified list as output. A QRPP may filter out recommendations, for example by removing recommendations for objects that the current user is not permitted to see, or may add or modify result metadata.

Recommendations are then presented to users via the activity graph task flows. For more information, see the "Adding the Activity Graph at Design Time" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

10.2 Configuration Roadmaps for Activity Graph

Use the roadmaps in this section as an administrator's guide through the configuration process:

- **Roadmap - Configuring the Activity Graph for WebCenter Portal**

  The flow chart (Figure 10–3) and table (Table 10–1) in this section provide an overview of the prerequisites and tasks required to get the activity graph working in WebCenter Portal.

Figure 10–3 Configuring the Activity Graph for WebCenter Portal

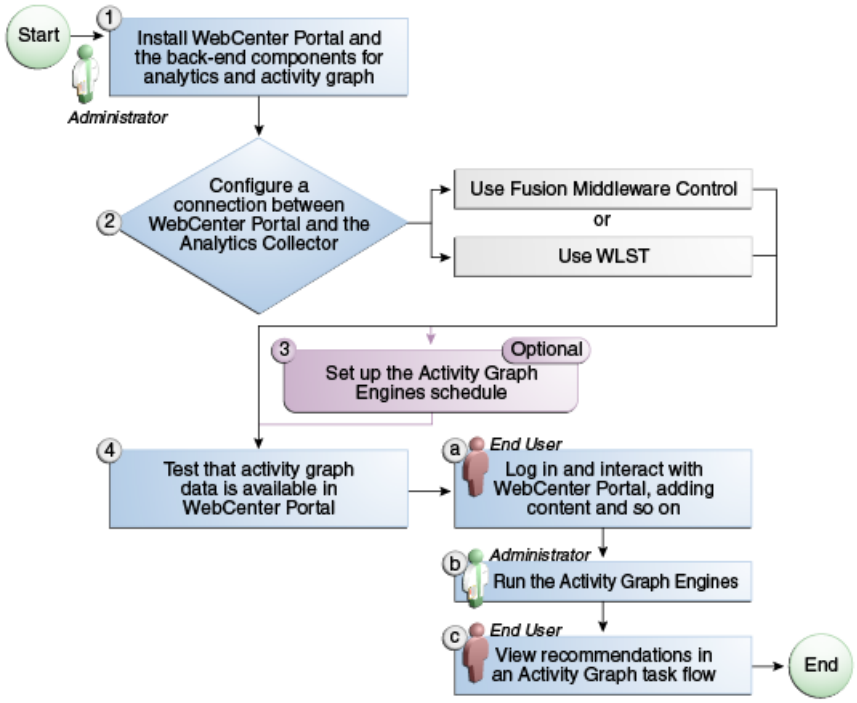


Table 10–1 Configuring the Activity Graph for WebCenter Portal

Actor	Task	Sub-task	Notes
Administrator	**1.** Install WebCenter Portal and the back-end components for analytics and the activity graph		
	2. Configure a connection between WebCenter Portal and the Analytics Collector using one of the following tools: ■ Fusion Middleware Control ■ WLST		
	3. (Optional) Set up the activity graph engines schedule		
End User/ Administrator	**4.** Test that activity graph data is available in the application	**4.a** Log in and interact with WebCenter Portal, for example, by adding content (End User) **4.b** Run the activity graph engines (Administrator) **4.c** View recommendations in an activity graph task flow, for example, the Recommended Connections task flow on the Profile page (End User)	

- **Roadmap - Configuring Activity Graph for a Portal Framework application**

  The flow chart (Figure 10–4) and table (Table 10–2) in this section provide an overview of the prerequisites and tasks required to get the activity graph working in Portal Framework applications.

Figure 10–4 Configuring Activity Graph for a Portal Framework application

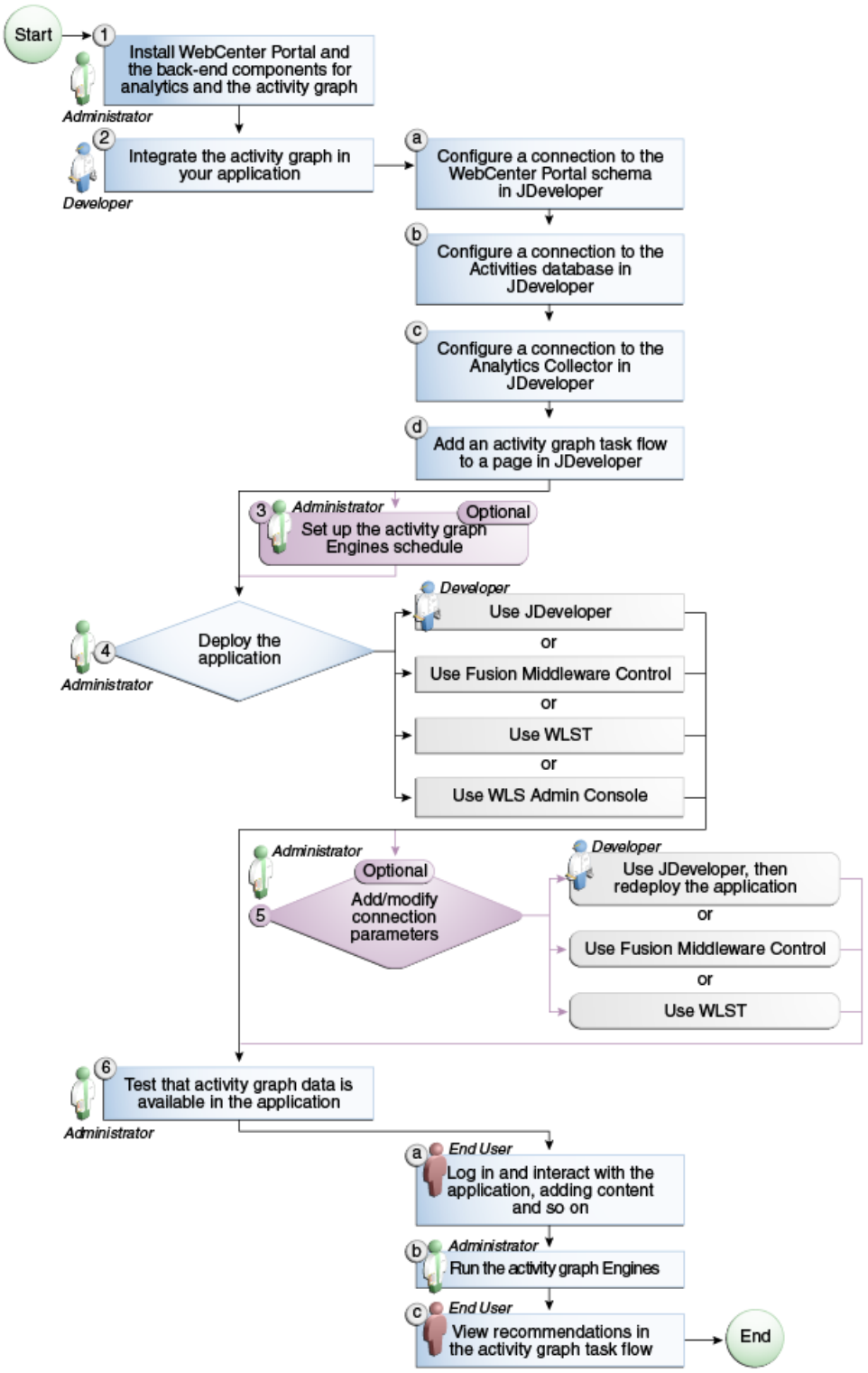


Table 10–2 Configuring Activity Graph for a Portal Framework applications

Actor	Task	Sub-task	Notes
Administrator	**1.** Install WebCenter Portal and the back-end components for analytics and the activity graph		
Developer	**2.** Integrate the activity graph in your application	**2.a** Configure a connection to the WebCenter Portal schema in JDeveloper **2.b** Configure a connection to the Activities database in JDeveloper **2.c** Configure a connection to the Analytics Collector in JDeveloper **2.d** Add an activity graph task flow to a page in JDeveloper	
Administrator	**3.** (Optional) Set up the activity graph engines schedule		
Developer/ Administrator	**4.** Deploy the application using one of the following tools: ■ JDeveloper (Developer) ■ Fusion Middleware Control (Administrator) ■ WLST (Administrator) ■ WebLogic Admin Console (Administrator)		
Developer/ Administrator	**5.** (Optional) Add/modify connection parameters using one of the following tools: ■ JDeveloper, then redeploy the application (Developer) ■ Fusion Middleware Control (Administrator) ■ WLST (Administrator)		
End User/ Administrator	**6.** Test that activity graph data is available in the application	**6.a** Log in and interact with the application, for example, by adding content (End User) **6.b** Run the activity graph engines (Administrator) **6.c** View recommendations in the activity graph task flow, for example, the Recommended Connections task flow on your Profile page (End User)	

10.3 Activity Graph Prerequisites

The activity graph requires that the activity graph engines application has been installed and configured. For more information, see the *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*.

In addition, in your application you must create a connection to the WebCenter Portal schema and to the Activities database. For more information, see the "Setting Up a Database Connection" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

The application must be configured to send usage events to the Analytics Event Collector. For more information, see Section 11.5, "Registering an Analytics Collector for Your Application."

Before the activity graph can make recommendations, the activity graph engines must have been run at least once to gather the data and calculate similarity scores. For more information see Section 10.4, "Preparing Data for the Activity Graph."

Additionally, the items suggested in the Similar Items task flow, the Top Items task flow, and the Recommendation data control depend on the tools and services available in your application. For example, documents are only recommended if the Documents tool is enabled. An item can also be filtered out of the recommendations by the Resource Authorizer of the tool/service that owns the item.

- A connection to a content repository is required to see documents, wikis, and blogs.
- A connection to a discussions server is required to see discussions.

In a cluster environment, all instances of the activity graph engines application should be disabled except for one. For more information, see the "Configuring Activity Graph for WebCenter Portal" section in *Oracle Fusion Middleware High Availability Guide*.

10.4 Preparing Data for the Activity Graph

The activity graph engines consist of three separate engines for gathering data, calculating similarity scores, and calculating search rankings. These engines are:

- The Gathering Engine—gathers activities from the Analytics tables and other repositories via a set of registered activity providers.
- The Collaborative Filtering Engine (CFE)—calculates similarity scores on pairs of objects and stores them in the activity graph for later generation of similarity recommendations. It does this by performing a set of similarity calculations. Similarity calculations are objects that tell the Collaborative Filtering Engine how to calculate similarity scores on a given set of domain and background node classes. Each resulting similarity score is a number between 0 and 1 designating how similar two objects are to each other given a specific criterion. Similarity calculations are specified by the following properties: their domain and background classes, a distance function, and a relation combination.
- The Rank Engine—calculates a measure of importance of every node in the activity graph. These activity ranks can be stored in a search index and combined at query time with a query-dependent score to order search results.For more information, see Section 10.7, "Setting Up Activity Rank for Oracle Secure Enterprise Search." These scores are also useful in ordering context-free recommendations. For this reason, they are also stored in the Relation Store.

Before the activity graph can begin to recommend objects, these engines must be run at least once to gather the data and calculate similarity scores. After this initial run, the engines can be run on demand, or on a schedule to ensure that new activities are captured and analyzed.

This section includes the following topics:

- Section 10.4.1, "Running the Activity Graph Engines on a Schedule"
- Section 10.4.2, "Running the Activity Graph Engines on Demand"

10.4.1 Running the Activity Graph Engines on a Schedule

You can run the activity graph engines on a schedule to ensure that new activities are captured and analyzed on a regular basis. This is useful for applications with heavy traffic and frequently updated content.

To run the activity graph engines on a schedule:

1. Log in to Fusion Middleware Control and navigate to the home page for the Activity Graph application (`activitygraph-engines`).

   Use the Navigator to access the page, select either:

   - Application Deployments > activitygraph-engines

   - WebLogic Domain > wc_domain > WC_Utilities > activitygraph-engines

2. Under **Web Modules**, click the URL for the activity graph Schedule and Status Page and log in.

   Note: To access this page, you must be a member of the `Administrators` group.

   The activity graph Schedule and Status page does not support multibyte user names and passwords, so you must log in using an ASCII-only user name and password.

   Tip: You can access the activity graph Schedule and Status page directly by going to the following URL:

   `http://`*`host`*`:`*`port`*`/activitygraph-engines`

   where `http://`*`host`*`:`*`port`* is the URL for the `WC_Utilities` managed server.

3. On the activity graph Schedule and Status page, select **Run on a schedule**.
4. In the **Start on** field, enter the date on which you want the schedule to start.
5. In the **Run every** field, enter a value to determine how regularly the process occurs. For example, to run the process every day, enter `1` in the field. To run the process every other day, enter `2` in the field, and so on.
6. From the **at** dropdown list, select the time of day at which you want the process to start.
7. Click **Start**.

   The process will run on the date specified at the time selected, and then will continue to run as you have scheduled.

10.4.2 Running the Activity Graph Engines on Demand

If the data in your application is not likely to change very frequently, you can run the activity graph engines on demand as and when required. You can also use this option to run the activity graph engines on demand in between regularly scheduled runs.

To run the activity graph engines on demand:

1. Log in to Fusion Middleware Control and navigate to the home page for the Activity Graph application (`activitygraph-engines`).

Use the Navigator to access the page, select either:

- Application Deployments > activitygraph-engines

- WebLogic Domain > wc_domain > WC_Utilities > activitygraph-engines

2. Under **Web Modules**, click the URL for the activity graph Schedule and Status Page and log in.

   Note: To access this page, you must be a member of the `Administrators` group.

   The activity graph Schedule and Status page does not support multibyte user names and passwords, so you must log in using an ASCII-only user name and password.

   Tip: You can access the activity graph Schedule and Status page directly by going to the following URL:

   `http://host:port/activitygraph-engines`

   where `http://host:port` is the URL for the `WC_Utilities` managed server.

3. Select **Run once now**.
4. Select **Incremental Update** to update the tables with any activities that occurred since the last time the activity graph engines ran.

   Select **Full Rebuild** to delete all existing data and repopulate the tables with all activities. This may take some time.

5. Click **Start**.

   You can monitor the progress of the process in the Status section of the dialog.

6. Click **Stop** at any time to stop the process, if required.
7. If you want to return to a regular schedule after the on demand process has run, be sure to select **Run on a schedule**, check that the details are correct, and then click **Start** to resume the schedule.

10.5 Customizing Reason Strings for Similarity Calculations

Similarity calculations can have associated reason strings to help users understand why a particular recommendation was made. When a person or object is recommended, if the highest scoring related similarity calculation has an associated reason string, that string is displayed in the task flow. You can edit the reason strings provided for similarity calculations, or create additional strings.

Each similarity calculation can define two strings for each reason to provide singular and plural phrasing.

Reason strings can be customized with the following tokens:

- `{RECOMMENDED_ITEM}`—The name of the current recommended item.
- `{NUMBER_OF_ITEMS}`—The number of objects in common. This corresponds to the numerator of the component score.

- `{TOTAL_ITEMS}`—The total number of items in common. This corresponds to the denominator of the component score. The meaning depends on the similarity function associated with the top similarity URN.
- `{SIMILARITY_CALCULATION}`—The name of the top similarity calculation.

For example, the `user-connect` similarity calculation defines the following two reason strings:

```
reason-user-connect=You share {NUMBER_OF_ITEMS} connections with
{RECOMMENDED_ITEM}.

reason-user-connect=You share {NUMBER_OF_ITEMS} connection with
{RECOMMENDED_ITEM}.
```

To customize reason strings for similarity calculations:

1. Open the `UIBundle.properties` file.
2. Locate the reason string that you want to customize and edit it as required.
3. To create a new reason string, use the following format:

   `reason-`*`similarity-calculation`*`=`*`string`*

4. Save the `UIBundle.properties` file.

10.6 Managing Activity Graph Schema Customizations

WebCenter Portal provides out-of-the-box integration with the activity graph that includes metadata definitions for mapping WebCenter Portal event data from Analytics. This metadata is automatically loaded the first time the activity graph engines application starts.

You can extend activity graph metadata to change how actions are gathered from Analytics by manipulating XML files. To work with the metadata, you must first export the data to an XML file. After editing the XML files, you can then import the metadata back into the activity graph.

This section includes the following topics:

- Section 10.6.1, "Exporting Activity Graph Metadata"
- Section 10.6.2, "Exporting Provider Configuration"
- Section 10.6.3, "Importing Activity Graph Metadata"
- Section 10.6.4, "Deleting Activity Graph Metadata"
- Section 10.6.5, "Renaming Actions and Node Classes"

For information about the ways you can extend activity graph metadata, see the "Extending the Activity Graph" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

10.6.1 Exporting Activity Graph Metadata

Use the WLST command `exportAGMetadata` to export activity graph metadata definitions to an XML file. For command syntax and examples, see the "exportAGMetadata" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For example:

```
exportAGMetadata(appName='activitygraph-engines',
directoryPath='/scratch/monty', definitionFileName='activityGraphMetaData.xml',
includeProviderConfigurations=1)
```

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

10.6.2 Exporting Provider Configuration

Use the WLST command `exportAGProviderConfiguration` to export provider configuration metadata, for a given provider, to an activity graph metadata definition file. For command syntax and examples, see the "exportAGProviderConfiguration" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For example:

```
exportAGProviderConfiguration(appName='activitygraph-engines',
directoryPath='/scratch/monty',
definitionFileName='activityGraph-analytics-mappings.xml',
urn='oracle.webcenter.activitygraph.providers.analytics')
```

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

10.6.3 Importing Activity Graph Metadata

Use the WLST command `importAGMetadata` to import activity graph metadata definitions from an XML file. For command syntax and examples, see the "importAGMetadata" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

10.6.4 Deleting Activity Graph Metadata

Use the WLST command `deleteAllAGMetadata` to delete all the activity graph metadata that is defined for a WebCenter Portal application. You should use this command in conjunction with the WLST command `importAGMetadata` to completely reinstall activity graph metadata. For command syntax and examples, see the "deleteAllAGMetadata" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

> **Note:** Only use this command if you plan to import a new set of metadata.

You can delete metadata for individual activity graph objects using the WLST command indicated:

- Node class (`deleteAGNodeClass`)
- Action (`deleteAGAction`)
- Similarity calculation (`deleteAGSimilarityCalculation`)
- Rank calculation (`deleteAGRankCalculation`)
- Provider assignment (`deleteAGProviderAssignment`)

- QRPP (`deleteAGQRPPRegistration`)

> **Note:** These `delete` methods delete metadata from the schema. As a result of this, any associated data in the Activities database is removed the next time the activity graph engines are run.

For more information, see the "Activity Graph" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

10.6.5 Renaming Actions and Node Classes

Use the WLST command `renameAGNodeClass` to change the URN of a node class currently registered with activity graph. For command syntax and examples, see the "renameAGNodeClass" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

Use the WLST command `renameAGAction` to change the URN of an action currently registered with activity graph. For command syntax and examples, see the "renameAGAction" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

> **Note:** These commands do not delete any metadata associated with the affected node class or action

10.7 Setting Up Activity Rank for Oracle Secure Enterprise Search

Enterprise content contributed through WebCenter Portal and Fusion applications has a rich structure that lends itself to using the Markov Chain Analysis mathematical technique. This technique is used by the rank engine to produce an *activity rank* for each node that improves end user searches by producing more relevant result sets. This is achieved by introducing a measure of the importance of various objects into the Oracle Secure Enterprise Search (Oracle SES) search index. This importance can then be factored into how results are ordered in combination with more standard search criteria (term frequency, and so on). The determination of an object's importance is predicated on the history of users' interactions with that object.

With activity rank, the importance of a person depends on the number of items the person creates and edits; the importance of those items; the number of people who connect with the person; and the importance of those people. The importance of an item depends on the importance of its author; the number of people who view, tag, and edit the item; and the importance of those people.

The rank engine process can be divided into four phases:

- Gathering—The rank engine queries the analytics store for data about the connections between users and documents.
- Reshaping—The data is transformed into a matrix.
- Multiplying—The matrix is used to calculate activity ranks.
- Result storage—The activity ranks are stored in the Oracle SES search server so that they can be used at search time.

Candidates for activity rank are limited to an intersection of WebCenter Portal objects having Oracle SES crawler implementations, WebCenter Portal analytics instrumentation, and registration for activity graph rank calculation.

For an object to receive an activity rank that will affect search behavior, it must be eligible to have its event data collected from Analytics, a rank computation generated by activity graph, and an indexed entry in search with the value of the attribute `wc_serviceId` matching one of the classes registered in activity graph. The currently supported objects are:

- Users
- Documents
- Blog entries
- Wiki pages

> **Note:** Ranks are still calculated and used even in the case where the class of object is not searchable itself.

The range of actions considered for computing activity rank are defined in the `rankCalculation` specified in the `activityGraphMetaData.xml` file. The out-of-the-box declaration includes the following user actions:

```
<component actionURN="connect" weight="10.0"/>
<component actionURN="edit" weight="20.0" inverse="true"/>
<component actionURN="view-count" weight="1.0"/>
<component actionURN="create" weight="100.0" inverse="true"/>
<component actionURN="create" weight="100.0"/>
<component actionURN="edit-count" weight="20.0"/>
<component actionURN="download" weight="5.0"/>
<component actionURN="tag" weight="10.0"/>
<component actionURN="comment" weight="10.0"/>
```

From this you can see that the single action of viewing a document conveys significantly less importance (`weight="1.0"`) to that document than the act of creating (`weight="100.0"`) or tagging (`weight="10.0"`) the document.

Additionally, when the inverse attribute is set to true, a relationship from object-to-user is denoted. The effect of this relationship is to enable users to accrue authority from objects whose rank appreciates. For example, the author of a document (a `create` relationship) collects rank from that document as its rank appreciates from actions performed by other users on that document—tagging, viewing, downloading—which then amplifies the weight of the user's future actions.

When the activity graph rank engine completes its rank calculation for all of the affected objects, it sends a resulting set of identifiers with normalized ranks between 1 and 10 to a plug-in class, the `SesRankResultAcceptor`. This class simply pushes the ranks into the search index using the Oracle SES SOAP API. Once accepted by the SOAP API, the ranks, or *docscores* as they are known in Oracle SES, are immediately factored into the search ranking (providing the DocScore feature is fully enabled).

So, for two or more items within the same strata of a result set, those with higher docscores will receive higher search scores than they would otherwise, potentially raising them to a higher rank within that strata.

Before You Begin

Since activity rank works in conjunction with Oracle SES, you must make sure that Oracle SES is installed and configured correctly. Also, as activity rank only affects searchable items, the rank engine should be run after the SES crawler has finished a run. For more information, see Chapter 18, "Managing Oracle Secure Enterprise Search

in WebCenter Portal."

To configure activity rank for Oracle SES:

1. The rank engine expects to use docscore attributes with external names of `DOC_SCORE_1` for `ACTIVITY-RANK` and `DOC_SCORE_2` for `LIKE-RANK`. Therefore, you must perform a one-time mapping call to establish these fields by calling the stored procedure `eq_sdata.create_sdata_attribute`:

   ```
   exec eq_sdata.create_sdata_attribute('ACTIVITY-RANK');
   exec eq_sdata.create_sdata_attribute('LIKE-RANK');
   ```

   Note: These stored procedures must be invoked from the server hosting the Oracle SES instance.

2. You must also add entries for these attributes to the Oracle SES `ranking.xml` file to determine the weight that they carry.

   For *Oracle SES 11.1.2.2 only*, add the following:

   ```
   <ranking>
     <docscore-factor>
       <attribute-name>ACTIVITY-RANK</attribute-name>
       <column-name>DOC_SCORE_1</column-name>
       <weight>1.0</weight>
     </docscore-factor>
     <docscore-factor>
       <attribute-name>LIKE-RANK</attribute-name>
       <column-name>DOC_SCORE_2</column-name>
       <weight>1.0</weight>
     </docscore-factor>
   </ranking>
   ```

   For *all other versions of Oracle SES*, add the following:

   ```
   <ranking>
     <docscore-factor>
       <attribute-name>ACTIVITY-RANK</attribute-name>
       <column-name>DOC_SCORE_1</column-name>
       <weight>1.0</weight>
     </docscore-factor>
     <docscore-factor>
       <attribute-name>LIKE-RANK</attribute-name>
       <column-name>DOC_SCORE_2</column-name>
       <weight>1.0</weight>
     </docscore-factor>
     <docscore-factor>
       <attribute-name>Doc_Score</attribute-name>
       <weight>1.0</weight>
     </docscore-factor>
   </ranking>
   ```

   Tip: The `ranking.xml` file is located in the following directory:

   ```
   $SES_HOME/search/webapp/config
   ```

3. Restart the Oracle SES middle tier so that the changes take effect.

4. Use the WLST command `setAGProperty` to set the following activity graph properties:

```
oracle.webcenter.activitygraph.providers.datasources.ses.soap.admin.url
oracle.webcenter.activitygraph.providers.datasources.ses.soap.query.url
```

For example:

```
setAGProperty(appName='activitygraph-engines',
propertyName='oracle.webcenter.activitygraph.providers.datasources.ses.soap.adm
in.url',
propertyValue='http://seshostname:7777/search/api/admin/AdminService',
propertyType='String')

setAGProperty(appName='activitygraph-engines',
propertyName='oracle.webcenter.activitygraph.providers.datasources.ses.soap.que
ry.url',
propertyValue='http://seshostname:7777/search/query/OracleSearch',
propertyType='String')
```

Setting these properties enables the `SESRankResultAcceptor` to connect to the Oracle SES server and record ranks in the index.

5. Use the WLST command `setAGPasswordCredential` to set the user names and passwords to use to access the URLs defined by the two properties.

> **Note:** The credentials should match a user that has access to all searchable items.

For example:

```
setAGPasswordCredential(appName='activitygraph-engines',
propertyName='oracle.webcenter.activitygraph.providers.datasources.ses.soap.adm
in.credential',
userName='eqsys',
password='MyPassword1')

setAGPasswordCredential(appName='activitygraph-engines',
propertyName='oracle.webcenter.activitygraph.providers.datasources.ses.soap.que
ry.credential',
userName='orcladmin',
password='MyPassword1')
```

6. Restart the managed server on which the activity graph engines application is deployed (that is, the `WC_Utilities` managed server).

10.8 Troubleshooting Issues with Recommendations

This section provides information to assist you in troubleshooting problems you may encounter while using the activity graph.

> **Note:** The following troubleshooting solutions assume that the activity graph engines are deployed correctly, the `WC_Utilities` managed server is up and running, and the property `openusage enabled` is `true`.
>
> You should also ensure that you have read Section 10.3, "Activity Graph Prerequisites."

10.8.1 Troubleshooting the Activity Graph Engines Schedule and Status Page

Problem

The activity graph Schedule and Status page throws an error while running the activity graph engines

Solution

Basic verification in this case is to verify the deployment status of the activity graph engines from the WebLogic Console.

Problem

When the activity graph engines are started from the Schedule and Status page, the status of the engines is not reflected in the UI.

Solution

Check the activity graph logs to verify whether the engines are actually running or not. If the logs show that the engines are running, then the issue is only with the UI and it will not have any effect on the recommendations being displayed in the task flows. If the logs do not show any entries for the gathering/CFE engines, then there might be a problem with the event mapping file.

> **Tip:** The activity graph engines logs can be found in:
>
> *domain*/log/WC_Utilities.out
> *domain*/servers/WC_Utilities/logs/WC_Utilities-diagnotics.log

Problem

Cannot log in to the activity graph Schedule and Status page.

Solution

The activity graph Schedule and Status page does not support multibyte user names or passwords. Log in as an administrator with an ASCII-only user name and password.

11

Managing Analytics

This chapter describes how to configure and manage Analytics for WebCenter Portal and Portal Framework applications. Analytics enables you to display usage and performance metrics for these applications.

Always use Fusion Middleware Control or the WLST command-line tool to review and configure back-end services for WebCenter Portal and Portal Framework applications. Any configuration changes that you make postdeployment are stored in the MDS metadata store as customizations. See Section 1.3.5, "Oracle WebCenter Portal Configuration Considerations." Any changes that you make to *Analytics Collector* configuration are stored in the Analytics database.

> **Note:** Changes that you make to Analytics configuration through Fusion Middleware Control or using WLST are not dynamic so you must restart the managed server on which the Analytics Collector or portal application is deployed for your changes to take effect. See Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

This chapter includes the following sections:

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console and the `Administrator` role in the deployed application:

- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.
- **Portal Framework application**: `Administrator` role granted through the Administration Console.

For more information about roles and permissions, see Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

11.1 About Analytics in WebCenter Portal

Analytics allows WebCenter Portal administrators and business users to track and analyze portal usage. Analytics provides the following basic functionality:

- **Usage Tracking Metrics:** Analytics collects and reports metrics for common portal functions, including community, page, portlet, and document visits.
- **Behavior Tracking:** Users can analyze portal metrics to determine usage patterns, such as portal visit duration and usage over time.
- **User Profile Correlation:** Users can correlate metric information with user profile information. Usage tracking reports can be viewed and filtered by user profile data such as country, company, or state. For more details, see the "Query Options" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

An overview of Analytics components and ready-to-use task flows are described in the following sections:

- Section 11.1.1, "Analytics Components"
- Section 11.1.2, "Analytics Task Flows"

11.1.1 Analytics Components

Figure 11–1 illustrates components for Analytics in WebCenter Portal:

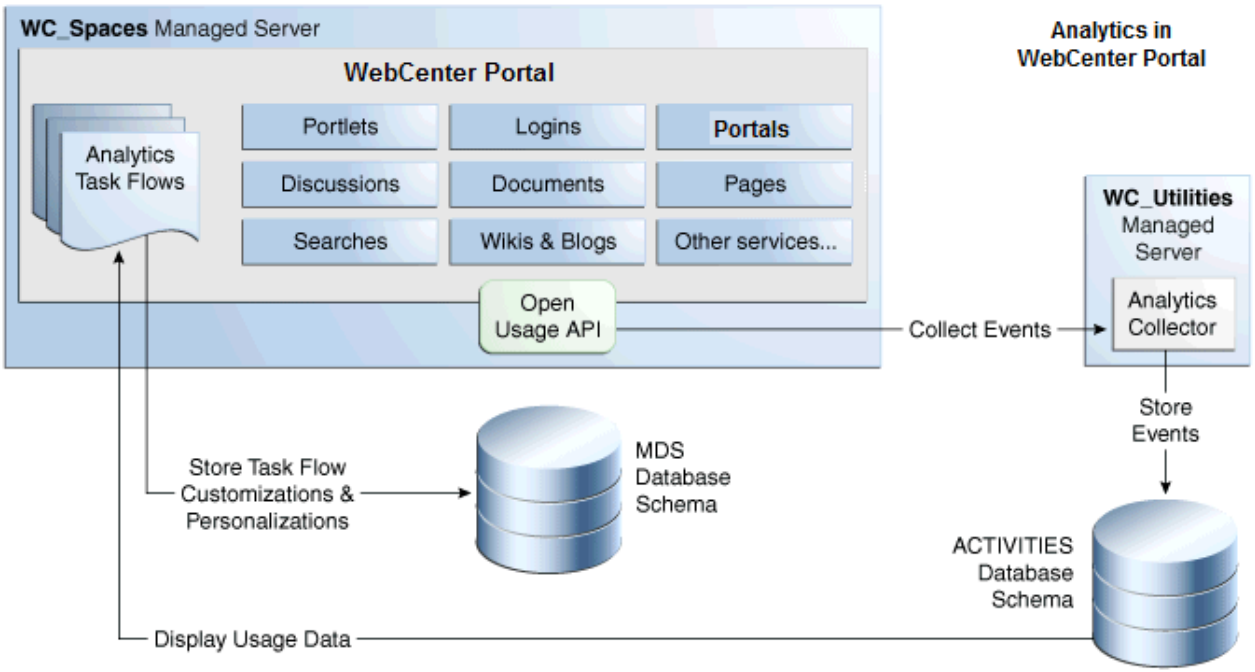
- **WC_Spaces** - The managed server in which the WebCenter Portal is deployed.

  (Portal Framework applications are deployed on different managed servers.)
- **WC_Utilities** - The managed server in which the Analytics Collector is deployed.
- **Event Data** - Analytics tracks and collects a defined set of events. A comprehensive set of the most common events are provided out-of-the-box.
- **Open Usage API** - The OpenUsage API sends metrics to the Analytics Collector using UDP (User Datagram Protocol).
- **Analytics Collector** - The Analytics Collector component gathers event data.

  Analytics Collectors can be clustered to provide increased scalability and reliability.
- **Analytics Database** - The Analytics database (ACTIVITIES) stores metrics gathered from portal and non-portal events.
- **Analytics Task Flows** - Analytics provides a series of task flows to report metrics for common portal functions.

- **MDS** - The Oracle Metadata Services (MDS) repository that stores task flow customizations.

Figure 11–1 Analytics Components



11.1.2 Analytics Task Flows

Table 11–1 lists the Analytics task flows available with WebCenter Portal. The task flows work similarly for Portal Framework applications and WebCenter Portal. For detailed information about these task flows and how to use them, see the "About Analytics"section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Table 11–1 Analytics Task Flows in WebCenter Portal

Analytics Task Flows	Description
WebCenter Portal Traffic	A summarized view for common events within the portal.
Page Traffic	Displays the number of page visits and the number of unique users that visited any page within the portal.
Login Metrics	Reports portal logins.
Portlet Traffic	Displays usage data for a portlet.
Portlet Response Time	Displays performance data for a portlet.
Portlet Instance Traffic	Displays usage data for a portlet instance. When the same portlet displays on several different pages, each placement is considered as a portlet instance.
Portlet Instance Response Time	Displays performance data for a portlet instance.
Search Metrics	Tracks portal searches.
Document Metrics	Tracks document views.
Wiki Metrics	Tracks most popular/least popular wikis.

Table 11–1 (Cont.) Analytics Task Flows in WebCenter Portal

Analytics Task Flows	Description
Blog Metrics	Tracks most popular/least popular blogs.
Discussion Metrics	Tracks most popular/least popular discussions.
Portal Traffic*	(WebCenter Portal only) Displays usage data for a portal.
Portal Response Time*	(WebCenter Portal only) Displays page performance data for a portal.

* These task flows are specific to WebCenter Portal. These task flows are not available for Portal Framework applications.

11.2 Configuration Roadmap for Analytics

The flow chart in Figure 11–2 and tasks in Table 11–2 provide an overview of the prerequisites and tasks required to get Analytics working in WebCenter Portal.

Figure 11–2 Configuring Analytics for Use in WebCenter Portal

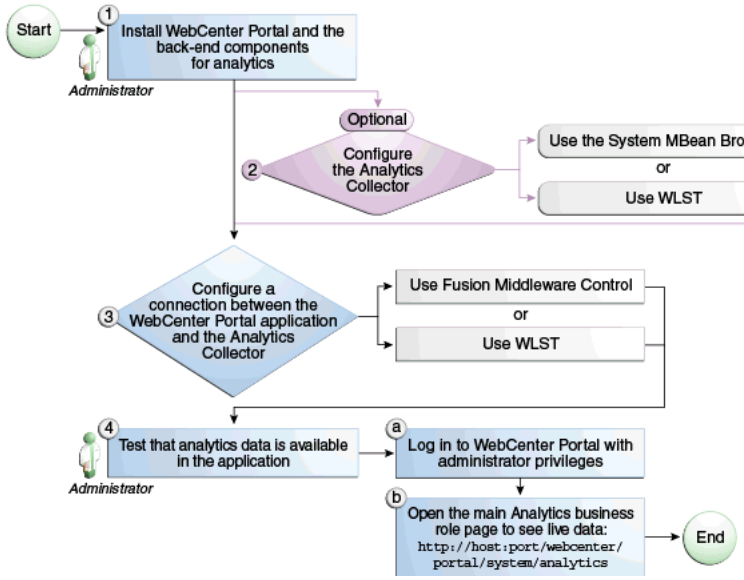


Table 11–2 Configuring Analytics for Use in WebCenter Portal

Actor	Task	Sub-task
Administrator	**1.** Install Oracle WebCenter Portal and the back-end components for Analytics.	
Administrator	**2.** (Optional) Configure the Analytics Collector Settings using either of the following tools: ▪ System MBean Browser ▪ WLST	

Table 11–2 (Cont.) Configuring Analytics for Use in WebCenter Portal

Actor	Task	Sub-task
Administrator	**3.** Configure a connection between the WebCenter Portal and the Analytics Collector using either of the following tools: ■ Fusion Middleware Control ■ WLST	
WebCenter Portal Administrator	**4.** Test that analytics data is available in WebCenter Portal	**4.a** Log into WebCenter Portal with administrator privileges **4.b** Open the main Analytics business role page to see live data: `http://host:port/webcenter/portal/system/Analytics`

11.3 Analytics Prerequisites

This section includes the following subsections:

- Section 11.3.1, "Analytics - Installation"
- Section 11.3.2, "Analytics - Configuration"
- Section 11.3.3, "Analytics - Security Considerations"
- Section 11.3.4, "Analytics - Limitations"

11.3.1 Analytics - Installation

The Analytics Collector is an optional installation option for Oracle WebCenter Portal. To install this product, select **Oracle WebCenter Portal Analytics Collector** in the Fusion Middleware Configuration Wizard. For detailed installation instructions, see *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*.

The Analytics schema (`ACTIVITIES`) and the WebCenter Portal schema (`WEBCENTER`) can be installed on the same database or on separate databases.

11.3.2 Analytics - Configuration

The Analytics Collector is configured to receive events out-of-the-box, using installation defaults. If the default values are not suitable for your installation or you have a cluster, you may configure different values using WLST or MBeans Browser. For more details, Section 11.4, "Configuring Analytics Collector Settings."

Out-of-the-box, WebCenter Portal is not configured to *send events* to the Analytics Collector. If you want to collect usage and performance metrics for WebCenter Portal (or any Portal Framework application) you must register the Analytics Collector and enable event collection. For more details, see Section 11.5, "Registering an Analytics Collector for Your Application." Once connected, analytics data is collected and displays in your application (through Analytics task flows) without further configuration.

11.3.3 Analytics - Security Considerations

In WebCenter Portal, Resource Catalogs only display Analytics task flows to users with appropriate permissions:

- Administrators - Users with the `Administrator` role have access to all Analytics task flows
- Moderators - Within a particular portal, members with the `Moderator` role have access to Analytics task flows that display usage data for that portal only

Analytics usage data is valuable for portal analysis but might be regarded as private or sensitive to portal users. To protect security and privacy interests associated with usage metrics WebCenter Portal administrators and individual portal moderators must manage page security such that only appropriate, specified users have access to pages that expose analytics data. See also, the "Setting Page Security" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Similarly, developers building Portal Framework applications must set up a suitable security model for exposing Analytics task flows and data. For details, see the "Setting up Security for Analytics Task Flows and Usage Data" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

11.3.4 Analytics - Limitations

Analytics task flows do not display custom event information.

11.4 Configuring Analytics Collector Settings

During installation, the Analytics Collector is configured to receive events using the following default values:

- **Collector Host Name** - localhost
- **Default Port** - 31314
- **Maximum Port Number** - 31314
- **Broadcast Type** - Multicast
- **Clustering** - The clustering settings do not apply. Clustering is not supported in this version.

If these default values are not suitable for your installation or you have a cluster, you can configure suitable values using WLST or the MBeans Browser in Fusion Middleware Control:

- Setting Analytics Collector Properties Using WLST
- Setting Analytics Collector Properties Using Fusion Middleware Control

These Analytics Collector configuration settings are stored in the Analytics database (`ACTIVITIES`).

11.4.1 Setting Analytics Collector Properties Using WLST

Use the WLST command `setAnalyticsCollectorConfig` to set event collection properties for the Analytics Collector. For command syntax and examples, see the "setAnalyticsCollectorConfig" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Note: To start using the property values you must restart the managed server on which the Analytics Collector application is deployed (`WC_Utilities`). For more information, see the "Starting and Stopping Managed Servers Using WLST" section in *Oracle Fusion Middleware Administrator's Guide*.

11.4.2 Setting Analytics Collector Properties Using Fusion Middleware Control

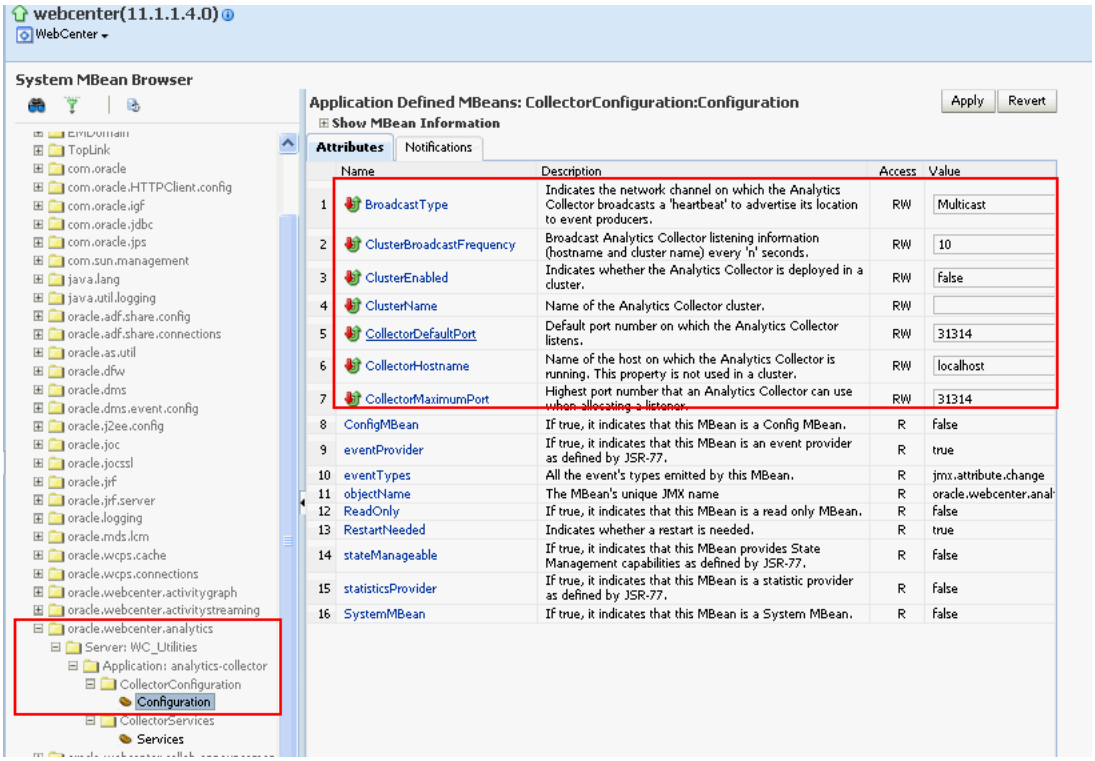
Use the Systems MBeans Browser in Fusion Middleware Control to set event collection properties for the Analytics Collector:

To configure the Analytics Collector (deployed on the `WC_Utilities` managed server):

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal or your Portal Framework application. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Open the System MBean Browser:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **System MBean Browser**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **System MBean Browser**.
3. Navigate to:
   Application Defined MBeans >oracle.webcenter.analytics >Server: WC_Utilities >Application: analytics-collector >CollectorConfiguration >Configuration

   Alternatively, search for `CollectorConfiguration` or filter the System MBean Browser tree using the MBean pattern: `oracle.webcenter.analytics:*`

Figure 11–3 System MBeans Browser - Analytics Collector Properties



4. Modify configuration properties for the Analytics Collector. For details, see Table 11–3.

Table 11–3 Analytics Collector - Configuration Properties

Field	Description
BroadcastType	Specify the network channel on which the Analytics Collector broadcasts a 'heartbeat' to advertise its location to event producers. Valid values are `Broadcast` and `Multicast`: **Broadcast** - use the standard network broadcast channel. **Multicast** - use a special fixed multicast address.
CollectorHostName	Enter the name of the host on which the Analytics Collector is running. The default setting is `localhost`.
CollectorDefaultPort	Enter the default port number on which the Analytics Collector listens. The default value is `31314`.
CollectorMaximumPort	Enter the highest port number that an Analytics Collector can use when allocating a listener. This property is mostly used in a clustered environment where multiple collectors run in the same box. Each collector listens for incoming UDP messages on a free port within a given port range. The range is from the default port number to the maxPort number.
ClusterEnabled	The clustering settings do not apply. Clustering is not supported in this version.
ClusterName	The clustering settings do not apply. Clustering is not supported in this version.

Table 11–3 (Cont.) Analytics Collector - Configuration Properties

Field	Description
HeartbeatFrequency	The clustering settings do not apply. Clustering is not supported in this version.

5. To start using the new settings you must restart the managed server on which the Analytics Collector application is deployed (`WC_Utilities`). For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

11.5 Registering an Analytics Collector for Your Application

Events raised in WebCenter Portal or a Portal Framework application using OpenUsage APIs can be sent to an Analytics Collector for use by Analytics, Recommendations and the Activity Graph Engine. If you intend to use any of the features or task flows provided by these tools you must connect WebCenter Portal or the Portal Framework application to an Analytics Collector.

While you can register multiple Analytics Collector connections for WebCenter Portal or your Portal Framework application, only one Analytics Collector is used - the default (or active) connection.

To start using a new configuration you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed.

This section includes the following subsections:

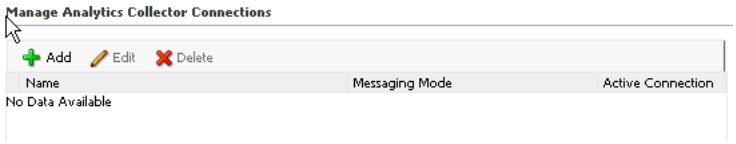
- Section 11.5.1, "Registering an Analytics Collector Using Fusion Middleware Control"
- Section 11.5.2, "Registering an Analytics Collector Using WLST"
- Section 11.5.3, "Disabling WebCenter Portal Event Collection"

11.5.1 Registering an Analytics Collector Using Fusion Middleware Control

To register an Analytics Collector for WebCenter Portal or a Portal Framework application:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal or your Portal Framework application. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Open the Service Configuration page:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.
3. From the list of services on the WebCenter Portal Service Configuration page, select **Analytics and Activity Graph**.
4. To connect to an Analytics Collector, click **Add** (Figure 11–4).

Figure 11–4 Configuring Analytics Collector Connections



5. Enter a unique name for this connection.

   The name must be unique (across all connection types) within WebCenter Portal or your Portal Framework application.

6. Select **Active Connection** to use this connection for Analytics and Activity Graph.

   While you can register multiple Analytics Collector connections for WebCenter Portal or your Portal Framework application, only one connection is used—the default (or active) connection.

7. Select **Enable WebCenter Portal Event Collection** to send analytics events raised using OpenUsage APIs to the Analytics Collector.

   Deselect this option if you do not want to collect analytics data.

8. Enter connection details for the Analytics Collector. For details, see Table 11–4.

Table 11–4 Analytics Collector Connection - Connection Details

Field	Description
Messaging Mode	This property specifies whether to send events to a clustered Analytics Collector in multicast mode or a single Analytics Collector using unicast communication. Clustering the Analytics Collector is not supported in the current release, so the only valid value for this release is `Unicast`.
Collector Host Name	If the messaging mode is set to `Unicast`, enter the host name where the Analytics Collector is running. The default setting is `localhost`.
Collector Port	Enter the port on which the Analytics Collector listens for events. The default value is `31314`.
Cluster Name	If the messaging mode is set to `Multicast`, enter the name of the cluster where a clustered Analytics Collector is running.
Timeout (Seconds)	If the messaging mode is set to `Multicast`, enter the length of time (in seconds) to wait for a response from the Analytics Collector. The default value is 30 seconds.

9. Click **OK** to save.

10. To start using the new (active) connection you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed. For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

11.5.2 Registering an Analytics Collector Using WLST

Use the WLST command `createAnalyticsCollectorConnection` to create an Analytics Collector connection for WebCenter Portal or your Portal Framework application. To update an existing connection, use `setAnalyticsCollectorConnection`. For command syntax and examples, see the

"createAnalyticsCollectorConnection" and "setAnalyticsCollectorConnection" sections in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Note: To start using the new connection, ensure that `isEnabled=1` and `default=1`, and then restart the managed server on which WebCenter Portal or your Portal Framework application is deployed. See the "Starting and Stopping Managed Servers Using WLST" section in *Oracle Fusion Middleware Administrator's Guide*.

11.5.3 Disabling WebCenter Portal Event Collection

If you do not want to collect events raised using OpenUsage APIs, you can stop event transmission temporarily or permanently.

This section includes the following subsections:

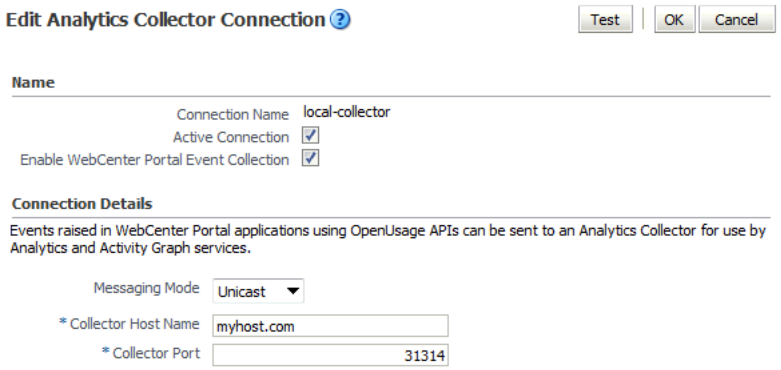
- Section 11.5.3.1, "Disabling WebCenter Portal Event Collection Using Fusion Middleware Control"
- Section 11.5.3.2, "Disabling WebCenter Portal Event Collection Using WLST"

11.5.3.1 Disabling WebCenter Portal Event Collection Using Fusion Middleware Control

To disable event collection for WebCenter Portal or your Portal Framework application:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal or your Portal Framework application. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Open the Service Configuration page:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.
3. From the list of services on the WebCenter Portal Service Configuration page, select **Analytics and Activity Graph**.
4. Select the connection in the table, and then click **Edit**.
5. Deselect **Enable WebCenter Portal Event Collection** (Figure 11–5).

Figure 11–5 Disabling Analytics Event Collection



6. To effect this change you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed. For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

11.5.3.2 Disabling WebCenter Portal Event Collection Using WLST

To disable event collection using WLST, run the `setAnalyticsCollectorConnection` command with the `isEnabled` argument set to `0 (false)`. For command syntax and examples, see the "setAnalyticsCollectorConnection"section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

11.6 Configuring User Profile Events Timing

User profile information is cached, meaning that changes to a user profile are not visible in reports until the cache is updated. The cache is limited to 1000 objects by default, with each object remaining in the cache for 60 minutes by default. You can change these values using WLST. To change the maximum number of objects in the cache, run the `setProfileCacheNumberOfObjects` command. To change the time an object remains idle in the cache, run the `setProfileCacheTimeToLive` command.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

11.7 Validating Analytic Event Collection

You can check whether events reach the Analytics Collector by checking the trace log at:

```
<base_domain_name>/servers/WC_Utilities/logs/analytics-collector/collector.trc
```

Event messages are similar to the following:

```
[2010-09-16T07:13:56.906-07:00] [WC_Utilities] [TRACE] []
[SRC_METHOD: OnMessageReceived] Event = [[
EVENT_TYPE: {http://www.myorg.com/videoapp}VIDEOVIEWS
VERSION: 3.0.XXXX
AS_DIMENSION_USER.USERID: testuser01
```

```
VIDEO.RESOURCEID: video8736
VIDEO.TITLE: Project Kick Off
VIDEO.LOOP: false
QUALITY: 720
PROPERTY_VERSION: 3.0.XXXX
```

To display analytics collector configuration information, enter the following URL:

`http://`*`hostname`*`:`*`WC_Utilities_port`*`/collector`

This page lists the following:

- Collector Default Port
- Collector Max Port
- Collector Server Name
- Broadcast Type
- Cluster Enabled
- Cluster Name
- Partitioning Enabled
- Time Dimension for This Year
- Space Dimension Exists (for WebCenter Portal)

11.8 Viewing the Current WebCenter Portal's Analytic Event List

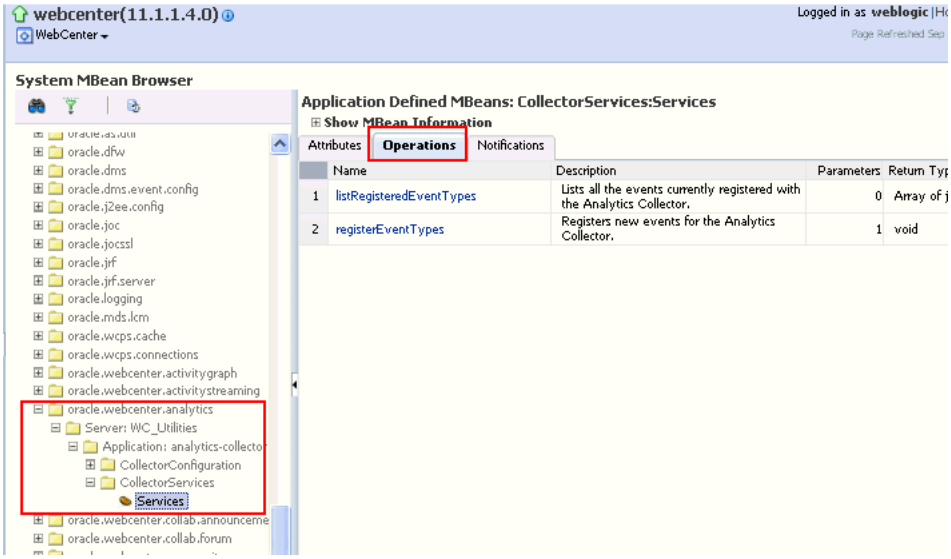
Use the Systems MBeans Browser in Fusion Middleware Control to see which events an Analytics Collector is configured to collect.

To display the current list of analytics events:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal or your Portal Framework application. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Open the System MBean Browser:
   - For the WebCenter Portal application - From the **WebCenter Portal** menu, select **System MBean Browser**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **System MBean Browser**.
3. Navigate to:
   Application Defined MBeans> oracle.webcenter.analytics Server: WC_Utilities> Application: analytics-collector> CollectorServices> Services

   Alternatively, search for `CollectorServices` or filter the System MBean Browser tree using the MBean pattern: `oracle.webcenter.analytics:*`
4. Select the **Operations** tab.

Figure 11–6 System MBeans Browser - Register Analytics Events



5. Click **listRegisteredEventTypes**.
6. Click **Invoke**.

Alternatively, use the WLST command `listAnalyticsEventTypes`. For command syntax and examples, see the "listAnalyticsEventTypes" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

11.9 Purging Analytics Data

For information about purging analytics data, see the "Purging Oracle WebCenter Portal's Analytics Data" section in *Oracle Fusion Middleware Administrator's Guide*.

11.10 Partitioning Analytics Data

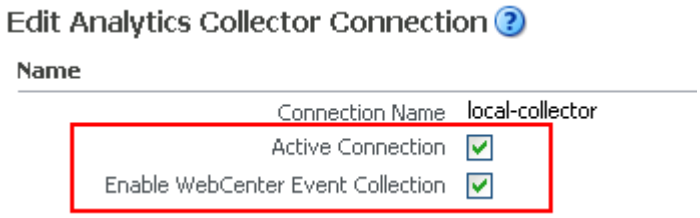
For information about partitioning analytics data, see the "Partitioning Oracle WebCenter Portal's Analytics Data"section in *Oracle Fusion Middleware Administrator's Guide*.

11.11 Troubleshooting Issues with Analytics

If users cannot see analytics in WebCenter Portal or your Portal Framework application, verify the following:

- Check that the Analytics Collector configuration is correct and in particular that both **Enable WebCenter Event Collection** and **Active Connection** are both set (Figure 11–7). See Registering an Analytics Collector for Your Application.

Figure 11–7 Enabling the Connection and Analytics Collection



If you make changes to the connection you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed. For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

- If WebCenter Portal or your Portal Framework application was recently upgraded, verify that the domain startup script does not contain legacy Analytics Collector settings as these values override any connection details that you specify through Fusion Middleware Control or using WLST.

  1. Shut down the managed server on which WebCenter Portal or your Portal Framework application is deployed.
  2. Edit the domain startup script `setDomainEnv` located at:

     UNIX: `DOMAIN_HOME/bin/setDomainEnv.sh`

     Windows: `DOMAIN_HOME\bin\setDomainEnv.cmd`
  3. Remove Analytics Collector settings.
  4. Restart the managed server.

12

Managing Announcements and Discussions

This chapter describes how to configure and manage announcements and discussions for WebCenter Portal. Both announcements and discussions use the same connection to WebCenter Portal's Discussion Server.

Unless otherwise documented, do not make configuration changes within WebCenter Portal's Discussion Server. Always use Fusion Middleware Control or the WLST command-line tool to review and configure back-end services for WebCenter Portal and Portal Framework applications. For troubleshooting tips with WebCenter Portal's Discussion Server, see Section 12.13, "Troubleshooting Issues with Announcements and Discussions."

Any configuration changes that you make postdeployment are stored in MDS metatdata store as customizations. See Section 1.3.5, "Oracle WebCenter Portal Configuration Considerations."

> **Note:** Configuration changes for discussions and announcements, through Fusion Middleware Control or using WLST, are not dynamic, so you must restart the managed server on which your application is deployed for changes to take effect. See Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

This chapter includes the following topics:

- Section 12.12, "Configuring Discussion Forum Options for WebCenter Portal"
- Section 12.13, "Troubleshooting Issues with Announcements and Discussions"

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console and the `Administrator` role in the deployed application:

- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.
- **Portal Framework application**: `Administrator` role granted through the Administration Console.

For more information about roles and permissions, see Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

12.1 About Discussions Server Connections

Announcements and discussions let users start, publish, and store discussions in WebCenter Portal. Users can create and expose announcements and discussions on the portal pages.

Discussions and announcements require a single connection to WebCenter Portal's Discussion Server. WebCenter Portal's discussion server can be installed with Oracle Fusion Middleware.

You can register additional discussion server connections through the Fusion Middleware Control Console or using WLST, but only one connection is active at a single time. See the following:

- Section 12.3.1, "Registering Discussions Servers Using Fusion Middleware Control"
- Section 12.3.2, "Registering Discussions Servers Using WLST"

WebCenter Portal

Some additional configuration is required to use discussions and announcements in WebCenter Portal. This includes choosing the category (on the discussions server) under which all WebCenter Portal discussions and announcements are stored, and more. This configuration takes place inside WebCenter Portal. For more information, see Section 12.12, "Configuring Discussion Forum Options for WebCenter Portal."

12.2 Discussions Server Prerequisites

This section includes the following subsections:

- Section 12.2.1, "Discussions Server - Installation"
- Section 12.2.2, "Discussions Server - Configuration"
- Section 12.2.3, "Discussions Server - Security Considerations"
- Section 12.2.4, "Discussions Server - Limitations"

12.2.1 Discussions Server - Installation

While installing WebCenter Portal, select to install WebCenter Portal's Discussion Server. Use the Repository Creation Utility (RCU) to create the `DISCUSSIONS` schema.

The Oracle Fusion Middleware Configuration Wizard automatically creates managed servers in the domain to host the selected WebCenter Portal components.

> **See Also:** *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*

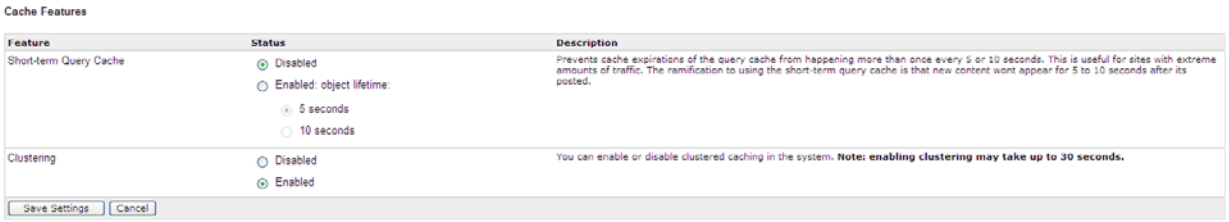
12.2.1.1 Discussions Server - High Availability Installation

> **Note:** Updates to discussion content do not refresh immediately when clustered caching is enabled. Users can click the **Refresh** icon to force a manual refresh at any time.

To set up WebCenter Portal's Discussion Server for high availability:

1. Install the `WC_Collaboration` domain in a clustered environment.
2. Log on to the discussions server admin console as an administrator by using the following URL format: `http://host:port/owc_discussions/admin`.
3. Go to the Cache Features page, and select to enable clustering (Figure 12–1).

Figure 12–1 Cache Features - Clustering



12.2.2 Discussions Server - Configuration

> **Note:** In a new or patched WebCenter Portal instance, the assigned security policy configuration is set to "no security policy." You must attach Oracle Web Services Manager (OWSM) security policies for the WebCenter Portal web service endpoint and the discussions authenticated web service endpoint.
>
> For detailed information, see Section 36.1.3.1, "Attaching Security Policies for WebCenter Portal and Discussions Web Service Endpoints."

There are numerous WLST commands for configuring the discussions server.

You can view, set, and remove WebCenter Portal's discussion server system properties with the following WLST commands, as described in Table 12–1:

Note: To execute discussions server WLST commands, such as `syncDiscussionServerPermissions`, the same user who connected to the admin server must also have administrative privileges on the discussions server.

For more information about WLST commands, see the "Discussions and Announcements" section in the "WebCenter Portal Custom WLST Commands" chapter in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

Table 12–1 Discussions Server WLST Commands

WLST Command	Purpose	More Information
`getDiscussionsServerProperty`	Return discussion server property values	See the "getDiscussionsServerProperty" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.
`setDiscussionsServerProperty`	Set discussion server properties	See the "setDiscussionsServerProperty" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.
`removeDiscussionsServerProperty`	Remove currently set discussion server property values	See the "removeDiscussionsServerProperty" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.
`addDiscussionsServerAdmin`	Grant system administrator permissions on the discussions server to a user or a group This command is useful when you connect the discussions server to a new identity store that does not contain any of the current administrators.	See the "addDiscussionsServerAdmin" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.
`syncDiscussionServerPermissions`	(WebCenter Portal only) For subportals to inherit discussions server permission changes applied to a parent When you update permissions for discussions or announcements for a portal hierarchy, subportals do not automatically inherit the corresponding permission changes on the discussions server.	See the "syncDiscussionServerPermissions" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

12.2.3 Discussions Server - Security Considerations

- WS-Security establishes a trust relationship between WebCenter Portal and WebCenter Portal's Discussion Server so that WebCenter Portal can pass the user identity information to the discussions server without knowing the user's credentials.

  Configure OWSM WS-Security for WebCenter Portal's Discussion Server, depending on your topology, by following any of the following procedures:

  – Section 36.1.3, "Configuring the Discussions Server for a Simple Topology"

  – Section 36.2.3, "Configuring the Discussions Server for a Typical Topology"

  – Section 36.3.3, "Configuring the Discussions Server for a Complex Topology"

- WebCenter Portal's Discussion Server-specific web services messages sent by WebCenter Portal to the discussions server are not encrypted. For message confidentiality, access the discussions server URL over Secure Socket Layer (SSL) or protect the Web service end points with an OWSM policy. For more information, see Chapter 35, "Configuring SSL" and Chapter 36, "Configuring WS-Security."

- By default, WebCenter Portal's Discussion Server is configured to use the embedded LDAP identity store: All users in the embedded LDAP store can log on to the discussions server, and all users in the `Administrators` group have administrative privileges on the discussions server.

  For your production environment, you must reassociate the identity store with an external LDAP server, as described in Section 31.1, "Reassociating the Identity Store with an External LDAP Server." In addition, you must either move the system administrator account to the external LDAP (as described in Section 31.4, "Moving the Administrator Account to an External LDAP Server"), or if you choose not to move the administrator account, you must perform some additional steps to identify the new administrator account for the discussions server as described in Section 31.4.1, "Migrating the Discussions Server to Use an External LDAP."

- You can configure WebCenter Portal's Discussion Server to leverage single sign-on security using Oracle Access Manager, Oracle Single Sign-On, or SAML-based single sign-on.

  Note: Direct login to the discussions server is not supported after SSO is configured. Log in through the Oracle HTTP Server URL.

  For detailed information, see Chapter 33, "Configuring Single Sign-on." For additional discussions-specific configuration instructions for Oracle Access Manager (OAM), see Chapter 33.2.6.2, "Configuring the Discussions Server for SSO."

Note: If you set up SAML single sign-on, with WebCenter Portal as the source application and WebCenter Portal's Discussion Server as the destination application, then you can access WebCenter Portal's Discussion Server administration pages from WebCenter Portal as follows:

- **Administration** > **Tools and Services**

  See Section 12.12.1, "Accessing the Discussions Server Admin Console."

- ***Portal_Name*** > **Settings** > **Tools and Services**

However, because the administration pages of WebCenter Portal's Discussion Server do not participate in single sign-on, if you access the administration pages directly, you are required to log in to the discussions server again.

- If WebCenter Portal is not integrated with a single sign-on solution, then different login sessions are required for the `owc_discussion` user (`/owc_discussions`) and the `owc_discussion` admin user (`/owc_discussions/admin`).
- User Identity: User identity management is handled by authentication providers settings specified in Oracle WebLogic Server using custom JPS Auth Factory. To check that the correct auth factory is running, go to WebCenter Portal's Discussions Server admin console System Properties page and confirm the following property values:
  - `owc_discussions.setup.complete_11.1.1.2.0=true`
  - `AuthFactory.className=oracle.jive.security.JpsAuthFactory`

    If the `AuthFactory.className` is set to this value, then set the `owc_discussions.setup.complete_11.1.1.2.0` property to `false` and restart WebCenter Portal's Discussion Server. This ensures that proper initialization is done for the application.

12.2.4 Discussions Server - Limitations

WebCenter Portal's Discussion Server URL supports only English and Spanish languages for displaying labels; however, data can be entered in UTF-8 format. Oracle recommends using WebCenter Portal (with all supported languages) for user operations in the discussions server. All WebCenter Portal-supported languages are supported for data, such as discussion topics or announcements, and they are displayed in the discussions server also.

Discussions and announcements do not support non-ASCII user names if the WebCenter Portal instance is running in a native encoding on Microsoft Windows. In a Linux environment, to allow support for non-ASCII user names in discussions and announcements, the server on which WebCenter Portal is deployed must have the environment variable `LC_ALL` set to `utf-8`.

WebCenter Portal

Do not change user permissions in the discussions server, as this might cause unexpected behavior. Always manage user permissions for discussions and announcements in WebCenter Portal. For more information, see Section 43.4.2.2.2, "Discussion Server Role Mapping."

12.3 Registering Discussions Servers

You can register multiple discussions server connections for WebCenter Portal, but only one is active at a single time.

To start using the new (active) connection you must restart the managed server on which WebCenter Portal is deployed.

This section includes the following subsections:

- Section 12.3.1, "Registering Discussions Servers Using Fusion Middleware Control"
- Section 12.3.2, "Registering Discussions Servers Using WLST"

12.3.1 Registering Discussions Servers Using Fusion Middleware Control

To register a discussions server:

1. Log in to Fusion Middleware Control and navigate to the home page for the application.

   For more information, see:

   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"

2. Do one of the following:

   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.

3. On the WebCenter Portal Service Configuration page, select **Discussions and Announcements**.

4. To connect to a new discussions server, click **Add** (Figure 12–3).

Figure 12–2 Configuring Discussion and Announcement Connections

5. Enter a unique name for this connection, and indicate whether this connection is the active (or default) connection for WebCenter Portal (Table 12–2).

Table 12–2 Discussion and Announcement Connection - Name

Field	Description
Connection Name	Enter a unique name for the connection. The name must be unique (across all connection types) within WebCenter Portal.

Table 12–2 (Cont.) Discussion and Announcement Connection - Name

Field	Description
Active Connection	Select to use this connection for Discussions and Announcements in WebCenter Portal. While you can register multiple discussions server connections for an application, only one connection is used for discussion and announcement—the default (or active) connection.

6. Enter connection details for the discussions server. For details, see Table 12–3.

Table 12–3 Discussion and Announcement Connection - Connection Details

Field	Description
Server URL	Enter the URL of the discussions server hosting discussion forums and announcements. For example: `http://discuss-server.com:8890/owc_discussions`
Administrator User Name	Enter the user name of the discussions server administrator. This account is used by the Discussions and Announcements tool to perform administrative operations on behalf of WebCenter Portal users. In the WebCenter Portal application, this account is mostly used for managing portal-related discussions and announcements. It is not necessary for this user to be a `super admin`. However, the user must have administrative privileges on the current root category for WebCenter Portal, that is, the category (on the discussions server) under which all portal-related discussions and announcements are stored. **Note:** If your application does not include portal-related functionality, then the administrator's user name is not required.

Table 12–3 (Cont.) Discussion and Announcement Connection - Connection Details

Field	Description
Authenticated User Web Service Policy URI	Select the policy this connection uses for authenticated access to the discussions server Web service. SAML (Security Assertion Markup Language) is an XML-based standard for passing security tokens defining authentication and authorization rights. An attesting entity (that already has a trust relationship with the receiver) vouches for the verification of the subject by a method called sender-vouches. The client policy specified must be compatible with the service policy that is configured for the `OWCDiscussionsServiceAuthenticated` endpoint in the discussions server. Out-of-the-box, the default *service policy* is WSS 1.0 SAML Token Service Policy (`oracle/wss10_saml_token_service_policy`). Options available are: ■ **WSS 1.0 SAML Token Client Policy** (oracle/wss10_saml_token_client_policy) ■ **WSS 1.1 SAML Token With Message Protection Client Policy** (oracle/wss11_saml_token_with_message_protection_client_policy) ■ **Global Policy Attachment** If your environment supports Global Policy Attachments, you must ensure that the default policy attached to the OWCDiscussionsServiceAuthenticated endpoint in the discussions server is set to `oracle/no_authentication_client_policy` using the WLST command `detachWebServicePolicy` or Enterprise Manager.
Public User Web Service Policy URI	Select the client policy this connection uses to enforce message security and integrity for public access to the discussions server Web service. The client policy specified must be compatible with the service policy that is configured for the `OWCDiscussionsServicePublic` endpoint in the discussions server. Out-of-the-box, a service policy is not configured for public access (None). Options available are: ■ **None** - This is the default setting. ■ **WSS 1.1 Message Protection Client Policy** (oracle/wss11_with_message_protection_client_policy) ■ **Global Policy Attachment** If your environment supports Global Policy Attachments, you must ensure that the default policy attached to the OWCDiscussionsServicePublic endpoint in the discussions server is set to `oracle/no_authentication_client_policy` using the WLST command `detachWebServicePolicy` or Enterprise Manager.
Recipient Key Alias	Enter the recipient key alias to be used for message protected policies (applicable to the `OWCDiscussionsServicePublic` and `OWCDiscussionsServiceAuthenticated` endpoints). This is the alias to the certificate that contains the public key of the discussions server in the configured keystore. See also Chapter 36, "Configuring WS-Security".

7. Configure advanced options for the discussion and announcement connection (Table 12–4).

Table 12–4 Discussion and Announcement Connection - Advanced Configuration

Field	Description
Connection Timeout (seconds)	Specify a suitable timeout for the connection. This is the length of time (in seconds) WebCenter Portal waits for a response from the discussions server before issuing a connection timeout message. The default is -1, which means that the service default is used. The service default is 10 seconds.

8. Sometimes, additional parameters are required to connect to the discussions server, for example, those listed in Table 12–5.

Table 12–5 Additional Discussion Connection Properties

Additional Connection Property	Description
application.root.category.id	(WebCenter Portal only) Application root category ID on the discussions server under which all discussion forums are stored. For example, if set to 3, then all forums are stored in the category with ID 3.
linkURL	URL used to link users to the discussions server's Admin Console. Only required if it is different to the **Server URL** property; for example, when SSO or HTTPS is configured. Use the following format to specify an alternative public external URL: `protocol://host:port` For example: `http://example.com:7777`

If additional parameters are required to connect to the discussions server, expand **Additional Properties** and enter details as required (Table 12–6).

Table 12–6 Discussion and Announcement Connection - Additional Properties

Field	Description
Add	Click **Add** to specify an additional connection parameter: ▪ **Property Name** - Enter the name of the connection property. ▪ **Property Value** - Enter the default value for the property. ▪ **Is Property Secured** - Indicate whether encryption is required. When selected, the property value is stored securely using encryption. For example, select this option to secure the `admin.password` property where the value is the actual password.
Delete	Click **Delete** to remove a selected property. Select the correct row before clicking **Delete**. **Note:** Deleted rows appear disabled until you click **OK**.

9. Click **OK** to save this connection.

10. To start using the new (active) connection, you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed.

For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

For WebCenter Portal, some additional configuration is recommended for the discussions. For details, see Section 12.12, "Configuring Discussion Forum Options for WebCenter Portal."

12.3.2 Registering Discussions Servers Using WLST

Use the WLST command `createDiscussionForumConnection` to create a discussions server connection. For command syntax and examples, see the "createDiscussionForumConnection" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

To configure discussions and announcements to actively use the new connection, set `default=true`.

Make sure to set additional properties for WS-Security. See Section 12.5.2, "Modifying Discussions Server Connection Details Using WLST."

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Note: To start using the new (active) connection, you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed. For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments.".

12.4 Choosing the Active Connection for Discussions and Announcements

You can register multiple discussions server connections for WebCenter Portal, but only one connection is active at a single time. The *active connection* becomes the back-end discussions server for:

- Discussions task flows (Discussion Forum Manager, Discussions, Popular Topics, Recent Topics, Watched Forums, Watched Topics)
- Announcements task flows (Announcements Manager, Announcements)

This section includes the following subsections:

- Section 12.4.1, "Choosing the Active Connection for Discussions and Announcements Using Fusion Middleware Control"
- Section 12.4.2, "Choosing the Active Discussion for Discussions and Announcements Using WLST"

12.4.1 Choosing the Active Connection for Discussions and Announcements Using Fusion Middleware Control

To change the active connection:

1. Log in to Fusion Middleware Control and navigate to the home page for the application.

   For more information, see:

- Section 6.2, "Navigating to the Home Page for WebCenter Portal"
- Section 6.3, "Navigating to the Home Page for Portal Framework Applications"

2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.
3. On the WebCenter Portal Services Configuration page, select **Discussions and Announcements**.

   The Manage Discussion and Announcement Connections table indicates the current active connection (if any).
4. Select the connection you want to make the active (or default) connection, and then click **Edit**.
5. Select the **Active Connection** check box.
6. Click **OK** to update the connection.
7. To start using the new (active) connection you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed.

   See Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

12.4.2 Choosing the Active Discussion for Discussions and Announcements Using WLST

Use the WLST command `setDiscussionForumConnection` with `default=1` to activate an existing connection. For command syntax and examples, see the "setDiscussionForumConnection" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

To disable a Discussions and Announcements connection, either delete it, make another connection the 'active connection', or use the `removeDiscussionForumServiceProperty` command:

```
removeDiscussionForumServiceProperty('appName='webcenter',
property='selected.connection')
```

Using this command, connection details are retained but the connection is no longer named as an active connection. For more information, see the "removeDiscussionForumServiceProperty" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Note: To start using the new (active) connection you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed. For more information see the "Starting and Stopping Managed Servers Using WLST" section in *Oracle Fusion Middleware Administrator's Guide*.

12.5 Modifying Discussions Server Connection Details

You can modify discussions server connection details at any time.

To start using the modified (active) connection you must restart the managed server on which the application is deployed.

This section includes the following subsections:

- Section 12.5.1, "Modifying Discussions Server Connection Details Using Fusion Middleware Control"
- Section 12.5.2, "Modifying Discussions Server Connection Details Using WLST"

12.5.1 Modifying Discussions Server Connection Details Using Fusion Middleware Control

To update connection details for a discussions server:

1. Log in to Fusion Middleware Control and navigate to the home page for the application.

   For more information, see:

   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"

2. Do one of the following:

   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.

3. On the WebCenter Portal Service Configuration page, select **Discussions and Announcements**.
4. Select the connection name, and click **Edit**.
5. Edit connection details, as required. For detailed parameter information, see Table 12–3 and Table 12–5.
6. Click **OK** to save your changes.
7. To start using the updated (active) connection you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed. See Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

12.5.2 Modifying Discussions Server Connection Details Using WLST

Use the WLST command `setDiscussionForumConnection` to edit connection details. For command syntax and examples, see the "setDiscussionForumConnection" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

To set additional parameters, use the `setDiscussionForumConnectionProperty` command. For more information, see the "setDiscussionForumConnectionProperty" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Note: To start using the updated (active) connection you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed. For more information, see the "Starting and Stopping Managed Servers Using WLST" section in *Oracle Fusion Middleware Administrator's Guide*.

12.6 Deleting Discussions Server Connections

You can delete discussions server connections at any time, but be careful when deleting the active connection. If you delete the active connection, none of the Discussions or Announcements task flows work, as they all require a back-end discussions server.

This section includes the following subsections:

- Section 12.6.1, "Deleting a Discussions Server Connection Using Fusion Middleware Control"
- Section 12.6.2, "Deleting a Discussions Server Connection Using WLST"

12.6.1 Deleting a Discussions Server Connection Using Fusion Middleware Control

To delete a discussions server connection:

1. Log in to Fusion Middleware Control and navigate to the home page for the application.

   For more information, see:

   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"

2. Do one of the following:

   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.

3. On the WebCenter Portal Services Configuration page, select **Discussions and Announcements**.

4. Select the connection name, and click **Delete**.

   Note: Before restarting the managed server, select another connection as active; otherwise, the discussions and announcements features are disabled.

5. To make this change you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed.

   For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

12.6.2 Deleting a Discussions Server Connection Using WLST

Use the WLST command `deleteConnection` to remove a connection. For command syntax and examples, see the "deleteConnection" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

Ensure that another connection is marked active; otherwise, the tool is disabled.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

> **Note:** To effect this change you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed. For more information, see the "Starting and Stopping Managed Servers Using WLST" section in *Oracle Fusion Middleware Administrator's Guide*.

12.7 Setting Up Discussions Defaults

Use the WLST command `setDiscussionForumServiceProperty` to set defaults for discussions in your application:

- `topics.fetch.size`: Maximum number of topics fetched by discussions and displayed in the topics view.
- `forums.fetch.size`: Maximum number of forums fetched by discussions and displayed in the forums view.
- `recentTopics.fetch.size`: Maximum number of topics fetched by discussions and displayed in the recent topics view.
- `watchedTopics.fetch.size`: Maximum number of topics fetched by discussions and displayed in the watched topics view.
- `watchedForums.fetch.size`: Maximum number of forums fetched by discussions and displayed in the watched forums view.
- `application.root.category.id`: Application root category ID on the discussions server under which all discussion forums are stored. For example, if set to `3`, then all forums are stored in the category with ID 3.
- `ForumGatewayManager.AUTO_START`: Communication through mail distribution lists can be published as discussion forum posts on a Discussions server, as described in the "Publishing Portal Mail in a Discussion Forum" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*. This parameter starts or stops the gateway for this communication.

  For WebCenter Portal, the default value is `1` (`true`), which means that as soon as you configure mail server settings through administration, the gateway starts. Set this to `0` (`false`), and restart the managed server, to stop the gateway and disable this feature.

  For Portal Framework applications, the default value is `0`. Set this to `1`, and restart the managed server, to start the gateway and enable this feature.

For command syntax and examples, see the "setDiscussionForumServiceProperty" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

12.8 Setting Up Announcements Defaults

Use the WLST command `setAnnouncementServiceProperty` to set defaults for announcements:

- `miniview.page_size`: Maximum number of announcements displayed in the Announcements quick view.
- `mainview.page_size`: Maximum number of announcements displayed in the Announcements main view.
- `linksview.page_size`: Maximum number of announcements displayed in the Announcements links view.
- `announcements.expiration.days`: Number of days that announcements display and remain editable.

For command syntax and examples, see the "setAnnouncementServiceProperty" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

12.9 Testing Discussions Server Connections

Try accessing the discussions server with the following URL:

`http://`*`host`*`:`*`port`*`/owc_discussions`

You should see a page listing all public information.

12.10 Granting Administrator Permissions on the Discussions Server

The WLST command `addDiscussionsServerAdmin` grants system administrator permissions on the discussions server to a user or a group. The WLST command `addDiscussionsCategoryAdmin` grants category administrator permissions on the discussions server to a user or a group for a specific category ID.

These commands are useful when you connect the discussions server to a new identity store that does not contain any of the current administrators.

For command syntax and examples, see the "addDiscussionsServerAdmin" and "addDiscussionsCategoryAdmin" sections in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

12.11 Granting Administrator Role on the Discussions Server

The default domain administrator created for WebCenter Portal is also the administrator for WebCenter Portal's Discussion Server. You can make a nondefault user the administrator for the discussions server too.

While creating a domain, if you specify any other user as the domain administrator, that user is granted all the domain administrative rights. However, after creating the domain, you must manually grant the administrator role to that nondefault user in both WebCenter Portal and the discussions server. For information on how to grant administrator privileges to a nondefault user for WebCenter Portal, see Section 32.6.1, "Granting the WebCenter Portal Administrator Role."

For WebCenter Portal's Discussion Server, the default user is the super administrator. This section describes how to grant administrator privileges to a nondefault user.

12.11.1 Granting the Discussions Server Administrator Role using WLST

The WLST command `addDiscussionsServerAdmin` lets you grant system administrator permissions on the discussions server to a user or a group. This is useful when you connect the discussions server to a new identity store. For command syntax and examples, see the "addDiscussionsServerAdmin" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

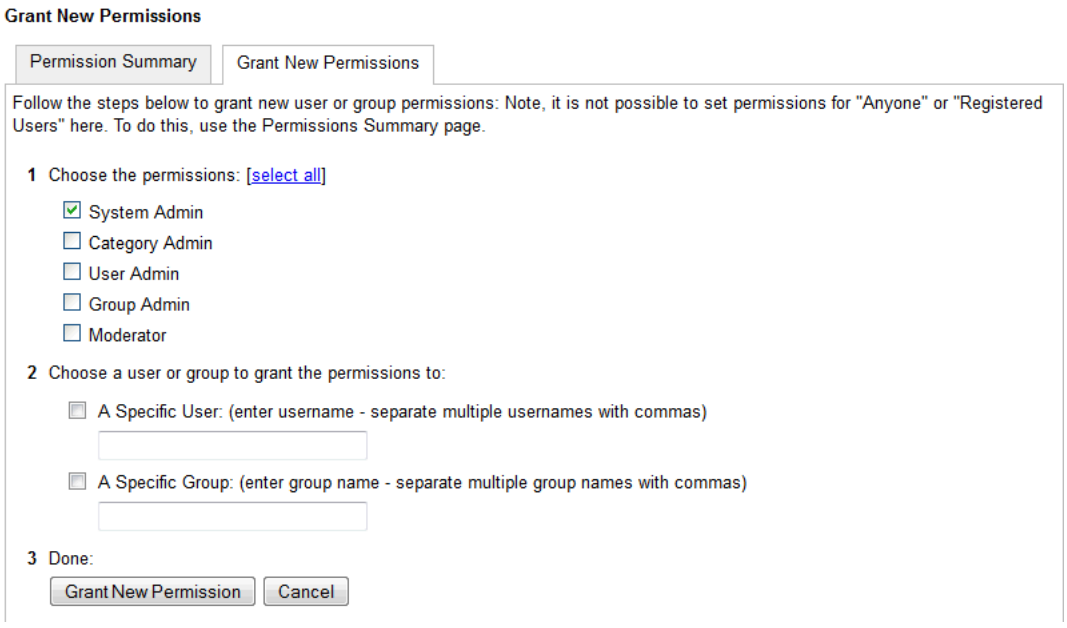
12.11.2 Granting the Discussions Server Administrator Role using the Admin Console

To grant the administrator role for WebCenter Portal's Discussion Server to a nondefault user:

1. Log on to the discussions server admin console as an administrator by using the following URL format: `http://host:port/owc_discussions/admin`.
2. Click the **Settings** link in the list of links across the top of the page.
3. Click the **Admins/Moderators** link, if not selected, in the navigation panel on the left.
4. On the **Admins & Moderators** page, click the **Grant New Permissions** tab.
5. Select the **System Admin** check box.
6. Select the **A Specific User** check box and specify the user to whom you want to grant administrative privilege for WebCenter Portal's Discussion Server.
7. Click **Grant New Permission**.

   You can now log on to WebCenter Portal's Discussion Server as the user whom you have assigned the administrative privilege.

Figure 12–3 Granting the Administrator Role on WebCenter Portal's Discussion Server



12.11.3 Revoking the Discussions Server Administrator Role

After assigning the discussions server administrator role to the required nondefault user, you may want to revoke the administrator role from the default user.

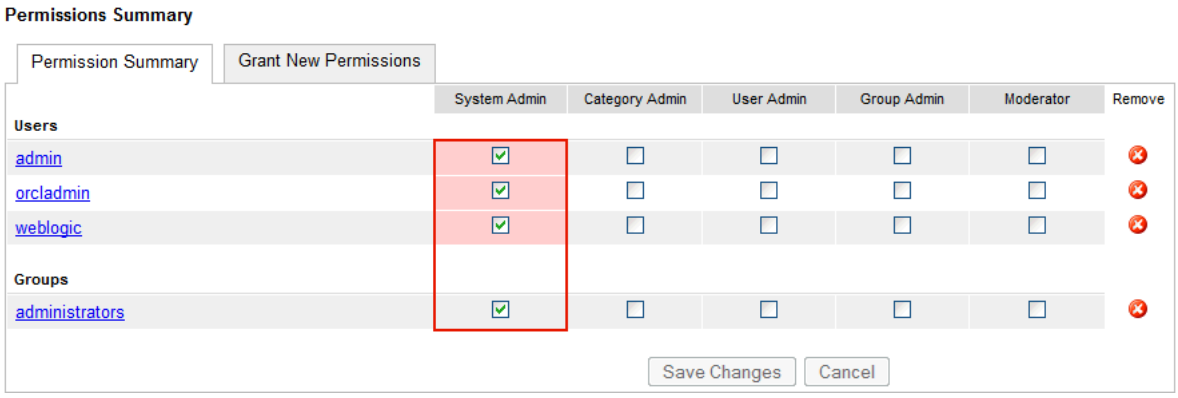
To revoke the administrator role:

1. Log on to discussions server admin console as the nondefault user whom you have assigned the administrator role.

   See also Section 12.12.1, "Accessing the Discussions Server Admin Console."

2. Click the **Settings** link in the list of links across the top of the page.
3. Click the **Admins/Moderators** link, if not selected, in the navigation panel on the left.
4. On the Admins & Moderators page, under the **Permission Summary** tab, uncheck the **System Admin** check box for the required user, for example, **weblogic**. (Figure 12–4)

Figure 12–4 Revoking the Administrator Role



5. Click **Save Changes**.

   The administrative privileges for managing WebCenter Portal's Discussion Server are now revoked from the default user.

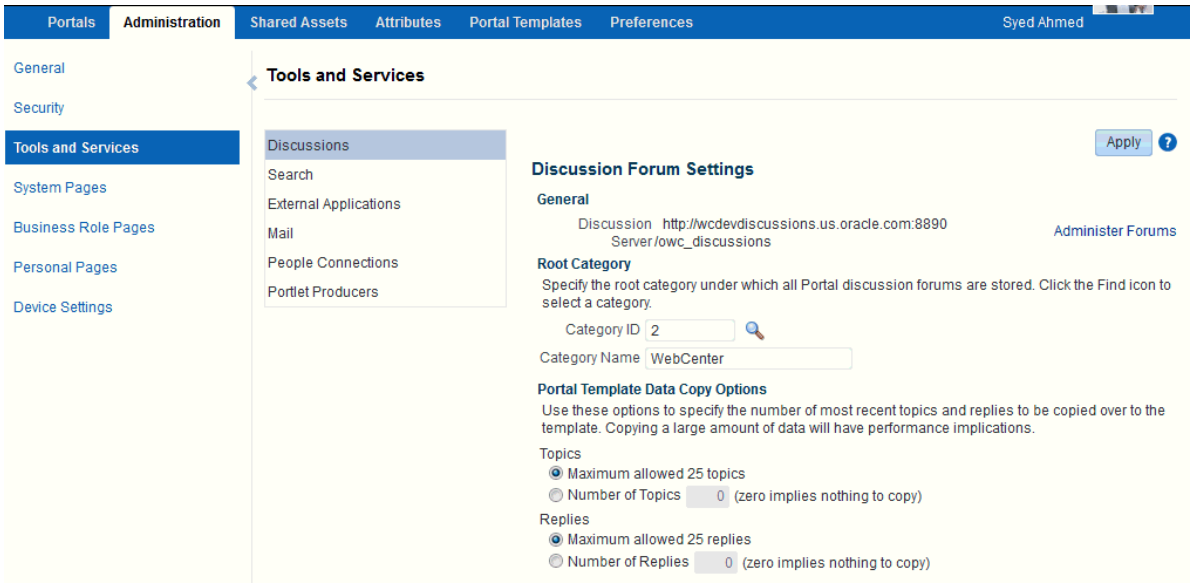
12.12 Configuring Discussion Forum Options for WebCenter Portal

Discussion forums allow members to capture, share, and preserve content that is relevant to their project or community goals.

> **Note:** To perform the tasks described in this section, you need WebCenter Portal `Administrator` (`Manage All`) permissions.

As an administrator, you are responsible for setting discussion forum options for the entire application through WebCenter Portal Administration pages (Figure 12–5).

Figure 12–5 Setting Discussion Forum Options



From the Discussions page you configure discussions-related setting, as well as access the discussions server administration pages:

- Section 12.12.1, "Accessing the Discussions Server Admin Console"
- Section 12.12.2, "Specifying Where Discussions and Announcements are Stored on the Discussions Server"
- Section 12.12.3, "Choosing How Many Discussion Topics to Save In Portal Templates"

> **Note:** The system administrator maintains the connection between WebCenter Portal and the discussions server. If you are experiencing issues with this connection, report the problem to the system administrator. See also Section 12.3, "Registering Discussions Servers."

12.12.1 Accessing the Discussions Server Admin Console

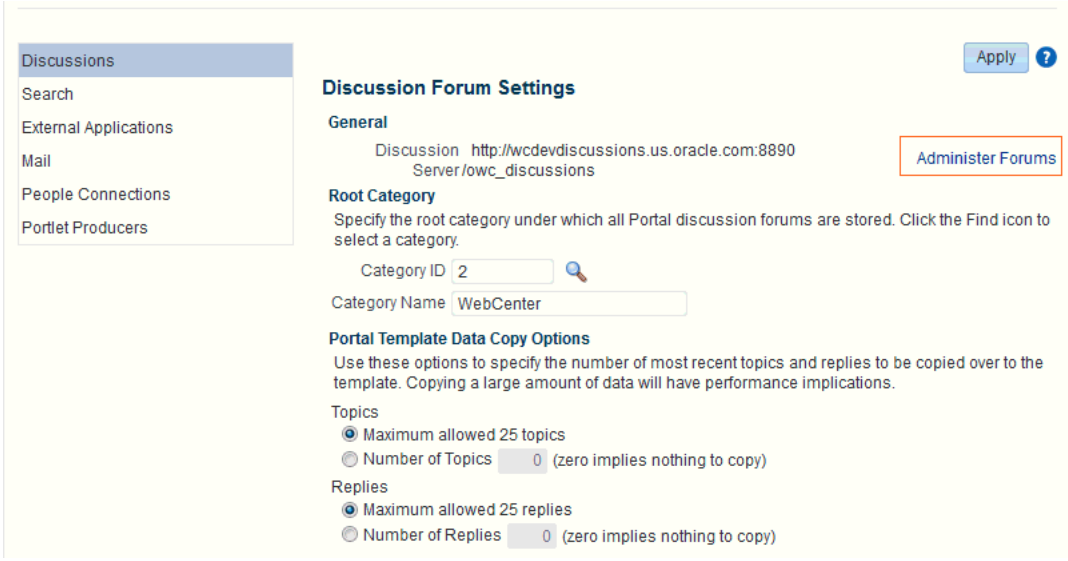
For convenience, you can access the discussions server's Admin Console, a web-based tool for configuring and managing discussion forums, from WebCenter Portal's Administration pages. In the discussions server's Admin Console, you can navigate all categories and forums and edit their properties, create new categories and forums, as well as set cache, security, and various other properties for the discussions server.

1. Open WebCenter Portal Administration pages.

   For details, see Section 47.2, "Accessing the Portal Builder Administration Page."

2. Click **Tools and Services**, and then select **Discussions**.
3. Click **Administer Forums** (Figure 12–6).

Figure 12–6 Administer Forums Link on the Discussion Forum Settings



4. Enter your discussions server administrator login credentials in the login page that appears.

> **Note:** If the **Forum Administration** link does not work, it could be because single sign-on or HTTPS is configured. Your system administrator must specify a public external URL (using the `linkURL` property).

12.12.2 Specifying Where Discussions and Announcements are Stored on the Discussions Server

WebCenter Portal administrators can change the root category (on the discussions server) under which all WebCenter Portal discussions and announcements are stored.

The default system root category is suitable in most cases but you can choose a different location. This might be useful when WebCenter Portal is connected to a discussions server that is hosting discussion forums for multiple applications.

Oracle recommendations:

- Choose a category that is dedicated to WebCenter Portal. There may be conflicts when multiple WebCenter Portals share the same root category.
- Do not switch the root category after WebCenter Portal is up and running. If you change the root category, then all the discussion forums under the old root continue to work, but you cannot create links to discussions or announcements stored in the old category.

You can retain existing discussions in a portal template saved with the data copy option. For example, in the WebCenter Portal Administration **Tools and Services - Discussions** page, enter the number (between 1 and 25) of most recent topics and replies to be copied over to the template.

Portal templates support single or multiple forums under the root category that you specify. With some templates, one forum is created automatically under the root category for each new portal based on that template.

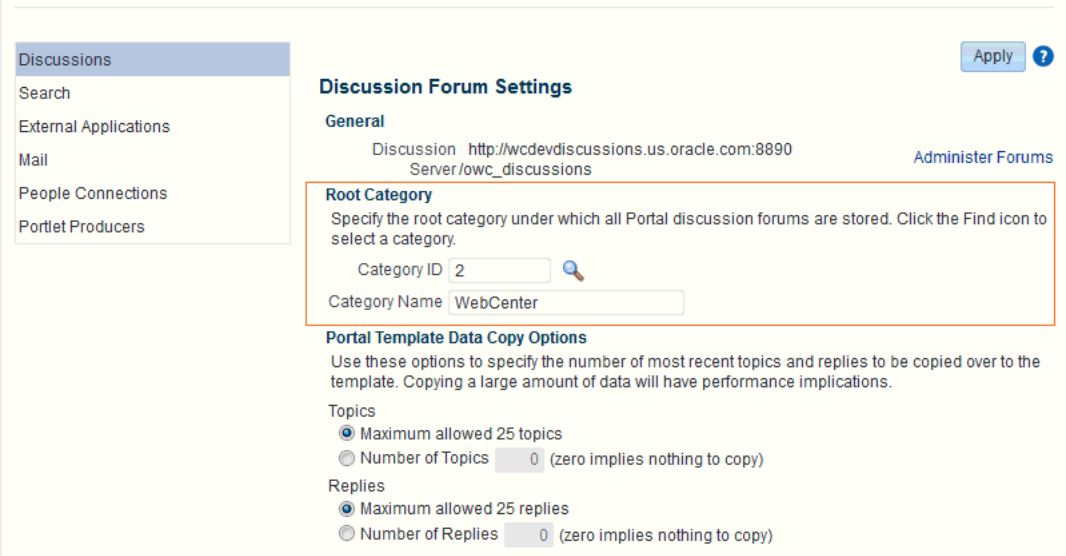
To specify where discussion forums are stored:

1. Open WebCenter Portal Administration.

   For details, see Section 47.2, "Accessing the Portal Builder Administration Page."

2. Click **Tools and Services**, and then select **Discussions** (Figure 12–7).

Figure 12–7 Specifying Where Discussions and Announcements are Stored



3. Specify an appropriate **Root Category** for storing discussions.

   Click the **Find** icon to view the categories available and then select the most appropriate location.

   To create a new category, click **Create Category**. You must have system administrator permissions on the discussions server to create new categories.

4. Click **Apply** to save the settings.

12.12.3 Choosing How Many Discussion Topics to Save In Portal Templates

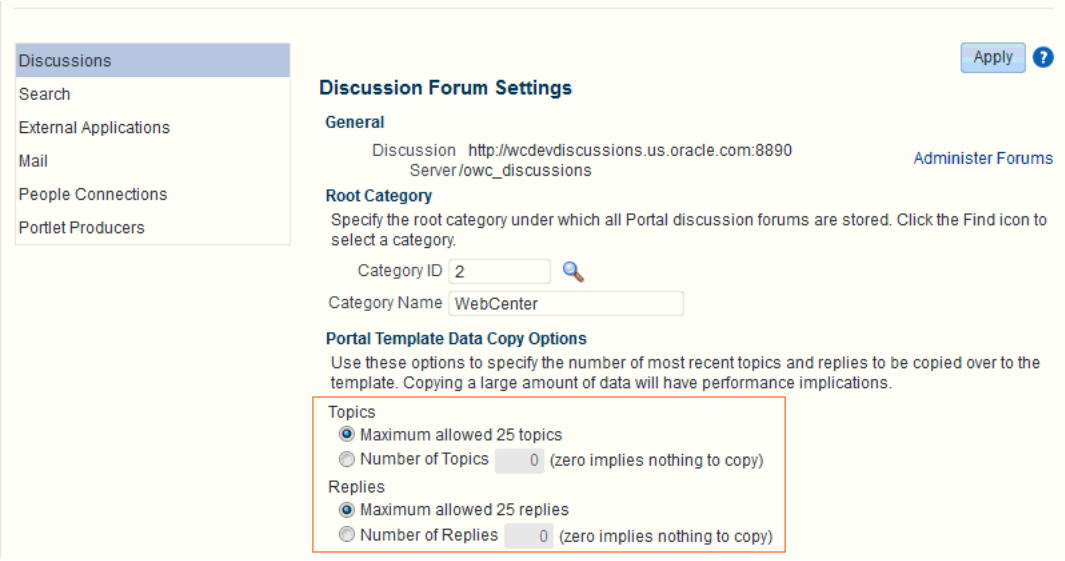
WebCenter Portal administrators can limit how many recent topics and replies are copied to portal templates. Because copying large amounts of data has performance implications, there is an upper limit of 25 topic or replies. If you prefer not to include any recent topics or replies in portal templates, specify zero.

1. Open WebCenter Portal Administration.

   For details, see Section 47.2, "Accessing the Portal Builder Administration Page."

2. Click **Tools and Services**, and then select **Discussions** (Figure 12–8).

Figure 12–8 Specifying the Number of Topics and Replies in a Portal



3. Specify an appropriate number of **Topics** and **Replies** to save in portal templates.
4. Click **Apply** to save the settings.

12.13 Troubleshooting Issues with Announcements and Discussions

This troubleshooting section includes the following subsections:

- Section 12.13.1, "Authentication Failed"
- Section 12.13.2, "Discussions Cannot Be Enabled in WebCenter Portal"
- Section 12.13.3, "Login Failed"
- Section 12.13.4, "Login Does Not Function Properly After Configuring Oracle Access Manager"
- Section 12.13.5, "Category Not Found Exceptions"
- Section 12.13.6, "Watched Topics and Recent Topics Not Displaying Topics From Multiple Discussion Forums"

12.13.1 Authentication Failed

Problem

WS-Security does not appear to be set properly for the connection between WebCenter Portal and the back-end discussions server. You may see the following error:

```
failure to authenticate the user WebLogic, due to: Authentication Failed
```

Solution

This error may be caused due to various reasons. Check the following:

- Ensure that the OWSM SAML policy setting is appropriately defined between the discussions connection and the discussions server.
- Review `WC_Spaces-diagnostic.log` for errors and exceptions relating to discussion services in WebCenter Portal. If the log does not provide enough

information to correct errors, then turn on debugging for the `oracle.webcenter.collab.share` and `oracle.webcenter.collab.forum` packages.

- For the discussions server, review `WC_Collaboration-diagnostics.log` and `jive.error.log` inside your domain's *`DOMAIN_HOME`*`/config/fmwconfig/servers/SERVER_NAME/owc_discussions/logs` directory. If the logs do not provide enough information to correct errors, then turn debugging on for the discussions server. To turn on debug logs, log on to the discussions server admin console, go to page logs, the Debug tab, and enable. Restart the WC_Collaboration managed server to change the logging setting.
- Make sure that time settings on WebCenter Portal and the back-end discussions server are in sync. This is important with OWSM WS-Security.

12.13.2 Discussions Cannot Be Enabled in WebCenter Portal

Problem

Discussions cannot be enabled in any portal in your WebCenter Portal installation.

Solution

This error may be caused due to various reasons. Check the following:

- The back-end discussions server is up and running and accessible. See Section 12.9, "Testing Discussions Server Connections."
- Administrator User Name (`adminUser`) property configured for the active connection has administrative privileges on the application root category (the category configured for WebCenter Portal). See Section 12.3, "Registering Discussions Servers."

  It is not necessary for this user to be a `super admin`. However, the user must have administrative privileges on the application root category configured for WebCenter Portal, that is, the category (on the discussions server) under which all WebCenter Portal discussions and announcement are stored.
- Application root category, where all WebCenter Portal discussions and announcements are stored, exists on the back-end discussions server.

  You can check the application root category ID configured for WebCenter Portal by navigating to WebCenter Portal **Administration**, then selecting **Tools and Services**, and then **Discussions**. See Section 12.12.2, "Specifying Where Discussions and Announcements are Stored on the Discussions Server."

12.13.3 Login Failed

Problem

You may see the following login exception:

```
caught exception running task oracle.webcenter.collab.share.LoginFailedException:
failure to authenticate the user monty, due to: Failed to read user monty from
database.
at
oracle.webcenter.collab.forum.internal.jive.JiveAuthenticator.login(JiveAuthentica
tor.java:213)
```

This occurs when an incorrect admin user name is specified.

Solution

Follow these steps:

1. Confirm that the admin user specified while creating the discussion forum connection has access to the Discussions Administration console at `http://host:port/owc_discussions/admin`.

   If the user does not have admin privileges, then use the WLST command `addDiscussionsServerAdmin` to provision the user. For more information, see Section 12.11.1, "Granting the Discussions Server Administrator Role using WLST."

2. Confirm that you have configured the discussion server with the appropriate `DISCUSSIONS` schema. If not, then create or extend the domain using `config.sh` or `was_config.sh`.

12.13.4 Login Does Not Function Properly After Configuring Oracle Access Manager

Problem

When you log in to WebCenter Portal's Discussion Server after configuring Oracle Access Manager single sign-on, a `500 - Internal Server Error` occurs.

Solution

1. If one does not exist, add a user as super admin on WebCenter Portal's Discussion Server using the WLST command `addDiscussionsServerAdmin`. For command syntax and examples, see the "addDiscussionsServerAdmin" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

2. Log on to the Discussions Admin Console with the super admin account, and navigate to System - System Properties.

   See Section 12.12.1, "Accessing the Discussions Server Admin Console."

3. Create or edit the property `owc_discussions.sso.mode`, and set its value to `true`.

   For more information, see Section 33.2.6.2, "Configuring the Discussions Server for SSO."

4. Restart WebCenter Portal's Discussion Server.

12.13.5 Category Not Found Exceptions

Problem

If you change the connection to use a different discussions server, and if you change WebCenter Portal's root category ID from **Administration** - **Tools and Services** - **Discussions**, then you could see exceptions like, "Category Not Found."

Solution

Restart the managed server on which WebCenter Portal is deployed.

12.13.6 Watched Topics and Recent Topics Not Displaying Topics From Multiple Discussion Forums

Problem

Portals created from the Discussion Site template include Recent Topics and Watched Topics task flows on the Home page. By default, both these task flows are configured to display information for a single forum. If your portal is configured to support multiple forums, topics from the other forums do not display in these task flows.

Solution

Edit the Watched Topics and Recent Topics task flows to remove the task flow parameters from the Forum ID field. In this instance, the Forum ID will be set to `${sessionContext['oracle.webcenter.collab.forum'].groupInfo[portalContext.currentPortalName].forumId}`. Delete this value and save the page.

12.13.7 Discussion and Announcement Updates Not Displayed

Problem

If clustered caching is enabled in your environment, content updates to discussions and announcements may not refresh immediately.

Solution

Click the **Refresh** icon to force a manual refresh at any time.

13

Managing Calendar Events

This chapter describes how to configure and manage events to expose personal Microsoft Exchange calendars in WebCenter Portal.

Always use Fusion Middleware Control or the WLST command-line tool to review and configure back-end services for WebCenter Portal and Portal Framework applications. Any configuration changes that you make, post deployment, are stored in the MDS metatdata store as customizations. See Section 1.3.5, "Oracle WebCenter Portal Configuration Considerations."

> **Note:** Configuration changes for events, through Fusion Middleware Control or using WLST, are not dynamic, so you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed for your changes to take effect. See Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

This chapter includes the following topics:

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console and the `Administrator` role in the deployed application:

- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.
- **Portal Framework application**: `Administrator` role granted through the Administration Console.

For more information about roles and permissions, see Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

13.1 About Events Connections

In WebCenter Portal, events provides portal calendars that you can use to schedule meetings, appointments, and any other type of team, project, or group occasion. Events also enables you to access your personal Microsoft Exchange calendar, where you can schedule events that are not related to a particular portal.

InPortal Framework applications, events provides access to the personal Microsoft Exchange calendars only.

Personal calendars are available through a Microsoft Exchange Server; therefore, a connection to that server is required. You can register the Microsoft Exchange Server connection through the Fusion Middleware Control Console or using WLST.

You must mark a connection as active for events to work. You can register additional Microsoft Exchange Server connections, but only one connection is active at a time.

To view personal events in WebCenter Portal, users must have an account on the Microsoft Exchange Server.

13.2 Configuration Roadmaps for Personal Events

Use the roadmaps in this section as a guide through the configuration process for providing access to personal events:

- **Roadmap - Configuring Personal Events for WebCenter Portal**

  The flow chart (Figure 13–1) and table (Table 13–1) in this section provide an overview of the prerequisites and tasks required for personal events to work in WebCenter Portal.

Figure 13–1 Configuring Personal Events for WebCenter Portal

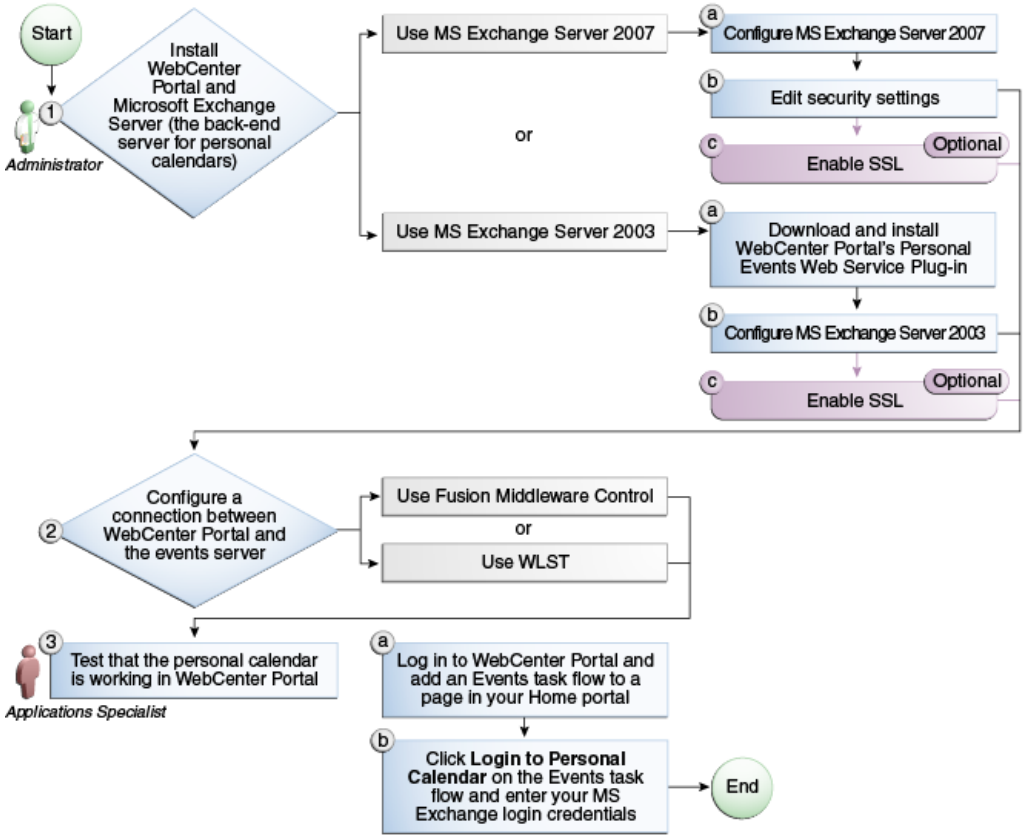


Table 13–1 Configuring the Personal Events for WebCenter Portal

Actor	Task	Sub-task	Notes
Administrator	**1.** Install WebCenter Portal and Microsoft Exchange Server		MS Exchange Server is the back-end component for personal calendars
	■ Install Microsoft Exchange Server 2007 (see Microsoft Exchange Server 2007 - Installation)	**1.a** Configure MS Exchange Server 2007 (see Microsoft Exchange Server 2007 - Configuration) **1.b** Edit security settings (see Microsoft Exchange Server 2007 - Security Considerations) **1.c** (Optional) Enable SSL (see Microsoft Exchange Server 2007 - Security Considerations)	
	■ Install MS Exchange Server 2003 (see Microsoft Exchange Server 2003 - Installation)	**1.a** Download and install WebCenter Portal's Personal Events Web Service Plug-in (see Microsoft Exchange Server 2003 - Configuration) **1.b** Configure MS Exchange Server 2003 (see Microsoft Exchange Server 2003 - Configuration) **1.c** (Optional) Enable SSL (see Microsoft Exchange Server 2003 - Security Considerations)	
	2. Configure a connection between the application and the events server using one of the following tools: ■ Fusion Middleware Control (see Registering Events Servers Using Fusion Middleware Control) ■ WLST (see Registering Event Servers Using WLST)		
End User	**3.** Test that the personal calendar is working in WebCenter Portal	**3.a** Log in to WebCenter Portal and add an Events task flow to a page in your Home portal **3.b** Click **Login to Personal Calendar** on the Events task flow and enter your MS Exchange Server login credentials	

- **Roadmap - Configuring Personal Events for Portal Framework applications**

  The flow chart (Figure 13–2) and table (Table 13–2) in this section provide an overview of the prerequisites and tasks required to get personal events working in Portal Framework applications.

Figure 13–2 Configuring Personal Events for Portal Framework Applications

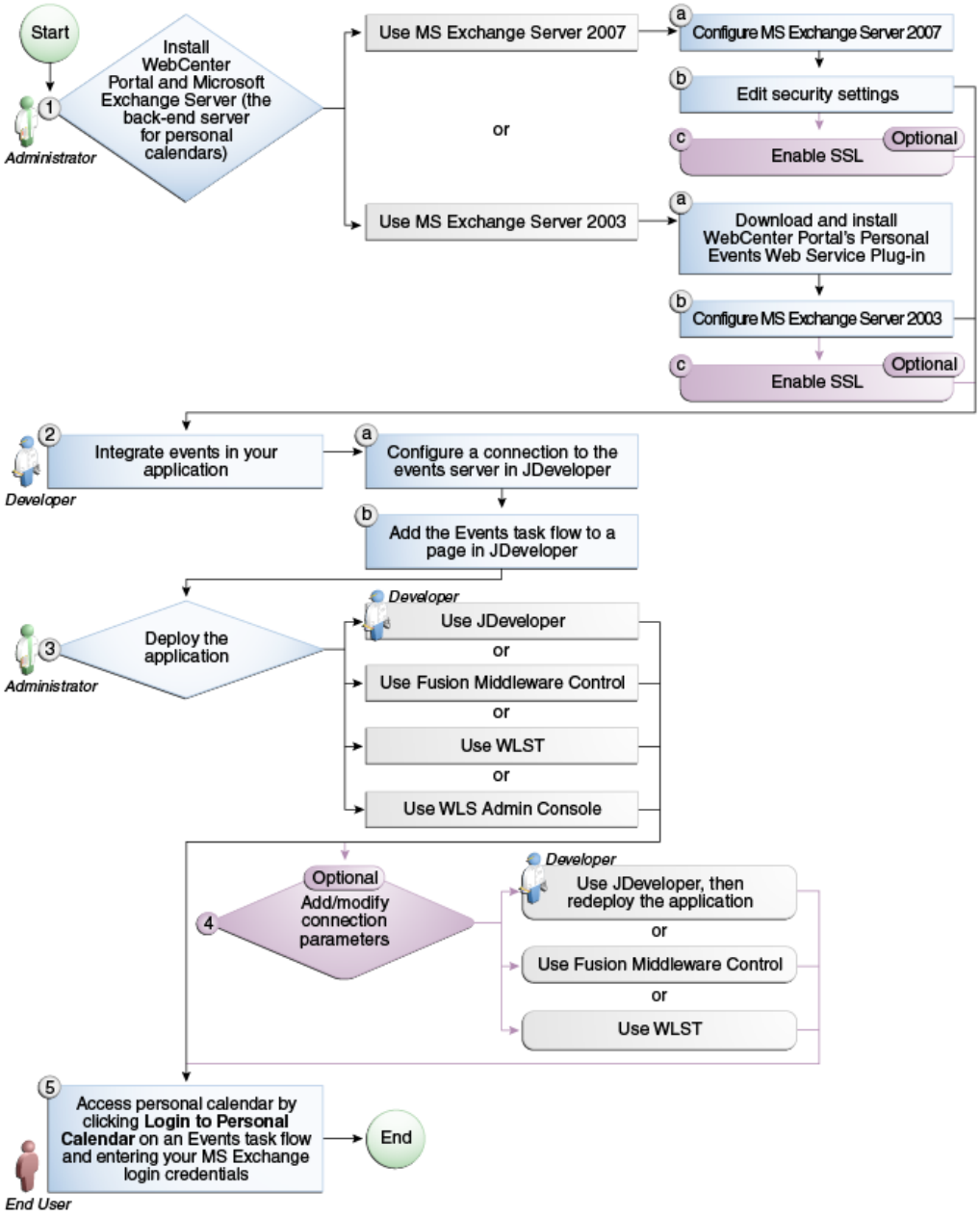


Table 13–2 Configuring Personal Events for Portal Framework Applications

Actor	Task	Sub-task	Notes
Administrator	**1.** Install WebCenter Portal and Microsoft Exchange Server		MS Exchange Server is the back-end component for personal calendars
	▪ Install Microsoft Exchange Server 2007 (see Microsoft Exchange Server 2007 - Installation)	**1.a** Configure MS Exchange Server 2007 (see Microsoft Exchange Server 2007 - Configuration) **1.b** Edit security settings (see Microsoft Exchange Server 2007 - Security Considerations) **1.c** (Optional) Enable SSL (see Microsoft Exchange Server 2007 - Security Considerations)	
	▪ Install MS Exchange Server 2003 (see Microsoft Exchange Server 2003 - Installation)	**1.a** Download and install WebCenter Portal's Personal Events Web Service Plug-in (see Microsoft Exchange Server 2003 - Configuration) **1.b** Configure MS Exchange Server 2003 (see Microsoft Exchange Server 2003 - Configuration) **1.c** (Optional) Enable SSL (see Microsoft Exchange Server 2003 - Security Considerations)	
Developer	**2.** Integrate Events in your application	**2.a** Configure a connection to the events server in JDeveloper **2.b** Add an Events task flow to a page in JDeveloper	
Developer/ Administrator	**3.** Deploy the application using one of the following tools: ▪ JDeveloper (Developer) ▪ Fusion Middleware Control (Administrator) ▪ WLST (Administrator) ▪ WLS Admin Console (Administrator)		
Developer/ Administrator	**4.** Add/modify connection parameters using one of the following tools: ▪ JDeveloper, then redeploy the application (Developer) ▪ Fusion Middleware Control (Administrator) ▪ WLST (Administrator)		
End User	**5.** Click **Login to Personal Calendar** on the Events task flow and enter your MS Exchange Server login credentials		

13.3 Events Prerequisites for Personal Events

This section includes the following subsections:

- Section 13.3.1, "Microsoft Exchange Server 2007 Prerequisites"
- Section 13.3.2, "Microsoft Exchange Server 2003 Prerequisites"

13.3.1 Microsoft Exchange Server 2007 Prerequisites

This section describes the Microsoft Exchange Server 2007 prerequisites when used as the server for personal events.

This section includes the following subsections:

- Section 13.3.1.1, "Microsoft Exchange Server 2007 - Installation"
- Section 13.3.1.2, "Microsoft Exchange Server 2007 - Configuration"
- Section 13.3.1.3, "Microsoft Exchange Server 2007 - Security Considerations"
- Section 13.3.1.4, "Microsoft Exchange Server 2007 - Limitations"

13.3.1.1 Microsoft Exchange Server 2007 - Installation

Refer to the Microsoft Exchange Server 2007 documentation for installation information.

13.3.1.2 Microsoft Exchange Server 2007 - Configuration

To use Microsoft Exchange Server 2007 as the server for personal events, you must edit the Microsoft Exchange Server 2007 web service WSDL to specify the location of the web service.

To specify the location of the Microsoft Exchange Server 2007 web service:

1. Open the WSDL file for the Microsoft Exchange Server web service.

   For example:

   ```
   C:\Program Files\Microsoft\Exchange
   Server\ClientAccess\exchweb\ews\Services.wsdl
   ```

2. Add a `service` section that points to your Microsoft Exchange Server web service.

   For example:

   ```
   <wsdl:definitions>
   ...
     <wsdl:service name="ExchangeServices">
       <wsdl:port name="ExchangeServicePort" binding="tns:ExchangeServiceBinding">
         <soap:address location="https://server.example.com/EWS/Exchange.asmx"/>
       </wsdl:port>
     </wsdl:service>
   </wsdl:definitions>
   ```

13.3.1.3 Microsoft Exchange Server 2007 - Security Considerations

Events includes a Microsoft Exchange Server 2007 adapter that communicates with the Microsoft Exchange Server 2007 generic web service through a JAX-WS proxy. To set up the communication between the adapter and the web service, you must edit the Microsoft Exchange Server security settings.

To edit security settings:

1. On the Microsoft Exchange Server, open Internet Information Services (IIS) Manager.
2. Under **Node** *computer_name* **> Web Sites >Default Web Site > EWS**, click **Properties**.
3. On the **Directory Security** tab, in the Authentication and access control, click **Edit**.
4. Select **Basic authentication**.
5. Click **OK**.

   You must enable anonymous access to `Services.wsdl`, `Messages.vsd`, and `Types.vsd` so that JAX-WS can access them to create the service port before committing any web service call.
6. Right-click **Services.wsdl** and select **Edit**.
7. On the **File Security** tab, in the Authentication and access control, click **Edit**.
8. Select **Enable anonymous access**.
9. Click **OK**.
10. Repeat steps 6 through 9 for **Messages.xsd** and **Types.xsd**.

Events uses Basic Authentication to communicate with the Microsoft Exchange Server. To secure the communication, you should enable SSL. For more information, see:

`http://www.microsoft.com/technet/prodtechnol/WindowsServer2003/Library/IIS/56bdf977-14f8-4867-9c51-34c346d48b04.mspx?mfr=true`

13.3.1.4 Microsoft Exchange Server 2007 - Limitations

There are currently no known limitations.

13.3.2 Microsoft Exchange Server 2003 Prerequisites

This section describes the Microsoft Exchange Server 2003 prerequisites when used as the server for personal events.

This section includes the following subsections:

- Section 13.3.2.1, "Microsoft Exchange Server 2003 - Installation"
- Section 13.3.2.2, "Microsoft Exchange Server 2003 - Configuration"
- Section 13.3.2.3, "Microsoft Exchange Server 2003 - Security Considerations"
- Section 13.3.2.4, "Microsoft Exchange Server 2003 - Limitations"

13.3.2.1 Microsoft Exchange Server 2003 - Installation

Refer to the Microsoft Exchange Server 2003 documentation for installation information.

13.3.2.2 Microsoft Exchange Server 2003 - Configuration

Microsoft Exchange Server 2003 does not provide a web service, so to use Microsoft Exchange Server 2003 as the server for events, you must install WebCenter Portal's Personal Events Web Service Plug-in on the IIS computer. The plug-in is available on the Oracle Fusion Middleware companion CD.

To install the Personal Events Web Service Plug-in:

1. Extract the contents of `ExchangeWebService.zip` to a folder within the Internet Information Services (IIS) server.

   You can find the ZIP file in the following directory on the Oracle Fusion Middleware companion CD:

   `/Disk1/WebCenter/services/cal/NT/ExchangeWebService.zip`

   Note: Make sure that the folder where you extract the file has the proper Read privileges. If necessary add Server Operators with additional Modify and Write privileges and Authenticated Users.

2. Open IIS Manager.
3. Under ***server_name*** **> Web Sites > Default Web Site**, create a new virtual directory called `ExchangeWS` (as the **Alias**).
4. Point the new virtual directory to the folder to which you extracted the ZIP file.
5. Make sure the folder has **Read** and **Run Scripts** privileges.
6. Right-click the new virtual directory and choose **Properties**.
7. On the **Virtual Directory** tab, under Application settings, from the **Execute permissions** dropdown list, select **Scripts and Executables**.
8. Click **Apply**.
9. On the **ASP.NET** tab, ensure that the **ASP.NET version** is **2.0.XXXXX**.

   Note: If ASP.NET is not available by default, then install the .NET 2.0 Framework from Microsoft.

10. Click **Edit Configuration**.
11. In the ASP .NET Configuration Settings dialog, make sure the **ExchangeServerURL** has the correct value.

    For example:

    `http://localhost:port/Exchange/User/calendar`

    Tip: The **ExchangeServerURL** is case-sensitive.

    Change the port, if necessary, to reflect the IIS port number. By default, this is 80.

12. Apply the changes and close the dialog.
13. Create a folder called `C:\WSErrorLogs`.
14. Test the web service from the IIS server and the WebCenter Portal server by accessing the following URL in your browser:

    `http://`*`host`*`:`*`port`*`/ExchangeWS/PersonalEventsWebService.asmx`

13.3.2.3 Microsoft Exchange Server 2003 - Security Considerations

Events uses Basic Authentication to communicate with the Microsoft Exchange Server. To secure the communication, you should enable SSL. For more information, see:

http://www.microsoft.com/technet/prodtechnol/WindowsServer2003/Library/IIS/56bdf977-14f8-4867-9c51-34c346d48b04.mspx?mfr=true

13.3.2.4 Microsoft Exchange Server 2003 - Limitations

There are currently no known limitations.

13.4 Registering Events Servers

You can register multiple events servers for WebCenter Portal, but only one is active at a single time.

To start using a new (active) connection you must restart the managed server on which the application is deployed.

This section includes the following subsections:

- Section 13.4.1, "Registering Events Servers Using Fusion Middleware Control"
- Section 13.4.2, "Registering Event Servers Using WLST"

13.4.1 Registering Events Servers Using Fusion Middleware Control

To register an events server:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal.

   For more information, see:

   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"

2. Do one of the following:

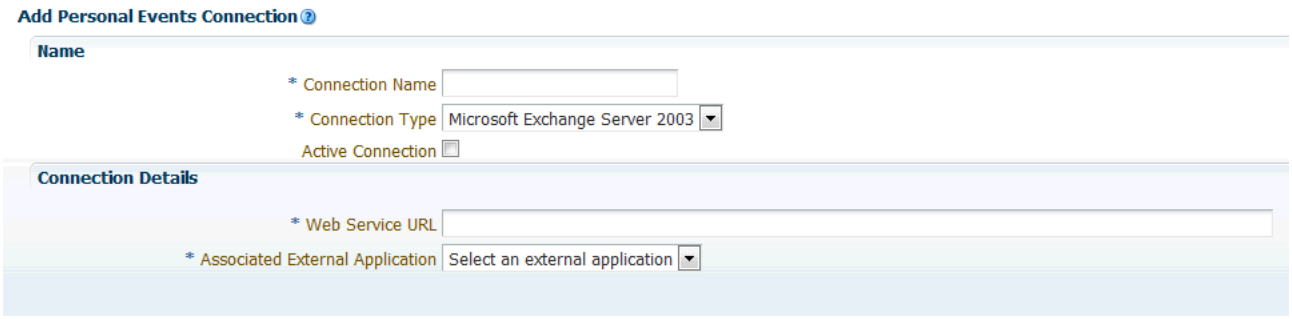
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal > Service Configuration**.

3. On the WebCenter Portal Service Configuration page, select **Personal Events**.

4. To connect to a new events server instance, click **Add**.

   The Add Personal Events Connection page appears (Figure 13–3).

Figure 13–3 Configuring Events Connections



5. Enter a unique name for this connection, specify the version of Microsoft Exchange Server, and indicate whether this connection is the active (or default) connection for WebCenter Portal (see Table 13–3).

Table 13–3 Personal Events Connection - Name

Field	Description
Connection Name	Enter a unique name for the connection. The name must be unique (across all connection types) within WebCenter Portal.
Connection Type	Select the Microsoft Exchange Server you want to connect to: ■ **Microsoft Exchange Server 2003** ■ **Microsoft Exchange Server 2007**
Active Connection	Select to use this connection for events in WebCenter Portal. While you can register multiple events server connections, only one connection is used by events—the default (or active) connection.

6. Enter connection details for the events server (Table 13–4).

Table 13–4 Personal Events - Connection Details

Field	Description

Web Service URL	Enter the URL of the web service exposing the event application.
	Use the format:
	protocol://*host*:*port*/*appWebServiceInterface*/*WSName*
	For example
	`http://myexchange.com:80/ExchangeWS/PersonalEventsWebService.asmx`
	`http://myexchange.com:80/EWS/Services.wsdl`

Associated External Application	Associate events with an external application. External application credential information is used to authenticate users against the Microsoft Exchange Server hosting events.

7. Click **OK** to save this connection.
8. To start using the new (active) connection you must restart the managed server on which WebCenter Portal is deployed.

   For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

13.4.2 Registering Event Servers Using WLST

Use the WLST command `createPersonalEventConnection` to create an events server connection. Use `setPersonalEventConnection` to alter an existing connection. For command syntax and examples, see the "createPersonalEventConnection" and "setPersonalEventConnection" sections in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Note: To start using the new (active) connection you must restart the managed server on which WebCenter Portal is deployed. See the "Starting and Stopping Managed Servers Using WLST" section in the *Oracle Fusion Middleware Administrator's Guide*.

13.5 Choosing the Active Events Server Connection

You can register multiple events server connections with WebCenter Portal, but only one connection is active at a time.

This section includes the following subsections:

- Section 13.5.1, "Choosing the Active Events Server Using Fusion Middleware Control"
- Section 13.5.2, "Choosing the Active Events Server Connection Using WLST"

13.5.1 Choosing the Active Events Server Using Fusion Middleware Control

To change the active connection:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal.

   For more information, see:

   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"

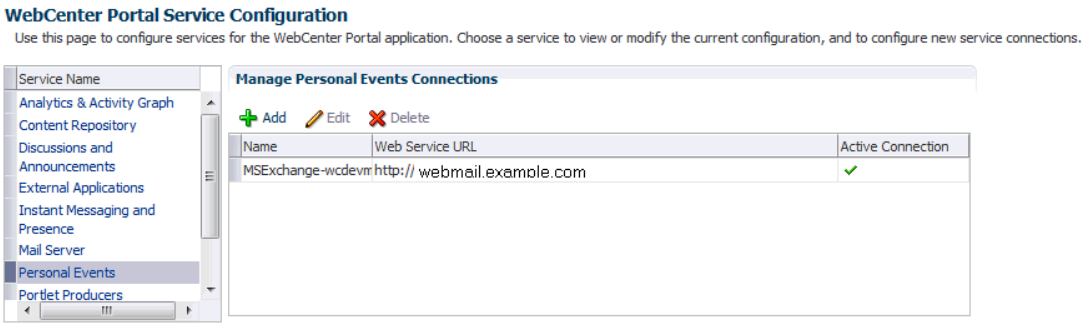
2. Do one of the following:

   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal > Service Configuration**.

3. On the WebCenter Portal Services Configuration page, select **Personal Events**.

   The Manage Personal Events Connections table indicates the current active connection, if any (Figure 13–4).

Figure 13–4 Active Connection for Personal Events



4. Select the connection you want to make the active (or default) connection, and then click **Edit**.
5. Select the **Active Connection** check box.
6. Click **OK** to update the connection.
7. To start using the new (active) connection you must restart the managed server on which WebCenter Portal is deployed.

For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

13.5.2 Choosing the Active Events Server Connection Using WLST

Use the WLST command `setPersonalEventConnection` with `default=1` (true) to activate an existing events server connection. For command syntax and examples, see the "setPersonalEventConnection" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

To subsequently disable an events connection, run the same WLST command with `default=0`. Connection details are retained but the connection is no longer named as an active connection.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Note: To start using the active connection you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed. For more information, see the "Starting and Stopping Managed Servers Using WLST" section in the *Oracle Fusion Middleware Administrator's Guide*.

13.6 Modifying Events Server Connection Details

You can modify events server connection details at any time.

To start using the updated (active) connection you must restart the managed server on which WebCenter Portal is deployed.

This section includes the following subsections:

- Section 13.6.1, "Modifying Events Server Connection Details Using Fusion Middleware Control"
- Section 13.6.2, "Modifying Events Server Connection Details Using WLST"

13.6.1 Modifying Events Server Connection Details Using Fusion Middleware Control

To update connection details for an events server:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal > Service Configuration**.
3. On the WebCenter Portal Service Configuration page, select **Personal Events**.
4. Select the connection name, and click **Edit**.
5. Edit connection details, as required.

For detailed parameter information, see Table 13–4

6. Click **OK** to save your changes.
7. To start using the updated (active) connection you must restart the managed server on which WebCenter Portal is deployed.

   For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

13.6.2 Modifying Events Server Connection Details Using WLST

Use the WLST command `setPersonalEventConnection` to edit an existing events server connection. For command syntax and examples, see the "setPersonalEventConnection" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Note: To start using the updated (active) connection you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed. For more information, see the "Starting and Stopping Managed Servers Using WLST" section in the *Oracle Fusion Middleware Administrator's Guide*.

13.7 Deleting Event Server Connections

You can delete events server connections at any time, but use caution when deleting the active connection. If you delete the active connection, users cannot create events in their personal calendar.

This section includes the following subsections:

- Section 13.7.1, "Deleting Event Server Connections Using Fusion Middleware Control"
- Section 13.7.2, "Deleting Event Server Connections Using WLST"

13.7.1 Deleting Event Server Connections Using Fusion Middleware Control

To delete an events server connection:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal.

   For more information, see:

   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"

2. Do one of the following:

   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal > Service Configuration**.

3. From the list on the WebCenter Portal Service Configuration page, select **Personal Events**.
4. Select the connection name, and click **Delete**.

> **Note:** Before restarting the managed server, select another connection as active; otherwise, the service is disabled.

5. To make this change you must restart the managed server on which WebCenter Portal is deployed.

   For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

13.7.2 Deleting Event Server Connections Using WLST

Use the WLST command `deleteConnection` to remove an events server connection. For command syntax and examples, see the "deleteConnection" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

> **Note:** To effect this change you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed. For more information, see the "Starting and Stopping Managed Servers Using WLST" section in the *Oracle Fusion Middleware Administrator's Guide*.

13.8 Testing Event Server Connections

To confirm the connection to the events server:

1. Add the Events task flow to a page in WebCenter Portal.

   > **Tip:** In WebCenter Portal, add the task flow to a page in your Home portal. See the "Adding an Events Task Flow to a Page" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click **Personal Events**, then click **Login to Personal Calendar**.
3. Enter your Microsoft Exchange Server login credentials.

   The personal events from your Microsoft Exchange Server should display in the task flow.

13.9 Troubleshooting Issues with Events

If users cannot see their personal events, verify the following:

- Is the Microsoft Exchange Server/IIS server accessible from the managed server on which WebCenter Portal is deployed? Can they ping each other?
- Is the configuration correct on the Microsoft Exchange Server? For more information, see Section 13.3.1.2, "Microsoft Exchange Server 2007 - Configuration" or Section 13.3.2.2, "Microsoft Exchange Server 2003 - Configuration."

- Is the events server connection correct in the managed server? For more information, see Section 13.4, "Registering Events Servers."
- Did the user enter the correct user name and password for the account on the Microsoft Exchange Server? The user name is usually an email address.

14

Managing Instant Messaging and Presence

This chapter describes how to configure and manage instant messaging and presence (IMP) for WebCenter Portal and Portal Framework applications.

Always use Fusion Middleware Control or WLST command-line tool to review and configure back-end tools and services for WebCenter Portal and Portal Framework applications. Any changes that you make to these applications, postdeployment, are stored in MDS metatdata store as customizations. See Section 1.3.5, "Oracle WebCenter Portal Configuration Considerations."

> **Note:** Configuration changes for instant messaging and presence, through Fusion Middleware Control or using WLST, are not dynamic, so you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed for changes to take effect. See Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

This chapter includes the following topics:

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console and the `Administrator` role in the deployed application:

- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.
- **Portal Framework application**: `Administrator` role granted through the Administration Console.

For more information about roles and permissions, see Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

14.1 About Instant Messaging and Presence Connections

Instant Messaging and Presence (IMP) lets you see the presence status of other authenticated application users (online, offline, busy, or away), and it provides quick access to interaction options, such as instant messages (IM) and mail.

A single connection to a back-end presence server is required. WebCenter Portal is certified with Microsoft Office Live Communications Server (LCS) 2005, Microsoft Office Communications Server (OCS) 2007, and Microsoft Lync 2010.

Notes: Oracle Beehive Server connections are not supported in this release.

You can register the presence server connection for your application through the Fusion Middleware Control Console or using WLST. You must mark a connection as active for IMP to work. You can register additional presence server connections, but only one connection is active at a time.

14.2 Instant Messaging and Presence Server Prerequisites

This section includes the following subsections:

- Section 14.2.1, "Microsoft Live Communications Server (LCS) Prerequisites"
- Section 14.2.2, "Microsoft Office Communications Server (OCS) Prerequisites"
- Section 14.2.3, "Microsoft Lync Prerequisites"

14.2.1 Microsoft Live Communications Server (LCS) Prerequisites

This section describes the Microsoft Live Communications Server 2005 (LCS) prerequisites as the presence server for instant messaging and presence.

This section includes the following subsections:

- Section 14.2.1.1, "Microsoft LCS - Installation"
- Section 14.2.1.2, "Microsoft LCS - Configuration"
- Section 14.2.1.3, "Microsoft LCS - Security Considerations"

14.2.1.1 Microsoft LCS - Installation

Refer to the Microsoft Live Communications Server 2005 documentation for installation information.

14.2.1.2 Microsoft LCS - Configuration

To use Microsoft Live Communications Server 2005 as the presence server for instant messaging and presence, you must install and configure the Microsoft RTC API v1.3, and you must install the Oracle RTC Web service for Microsoft LCS 2005.

1. To install the Microsoft RTC API v1.3, download the RTC SDK from Microsoft RTC Client API SDK 1.3, and run the installer. The installer provides the necessary installation components. If you choose the default options, the following two installers are available at `C:\Program Files\RTC Client API v1.3 SDK\INSTALLATION`:

   - `RtcApiSetup.msi`
   - `RtcSxSPolicies.msi`

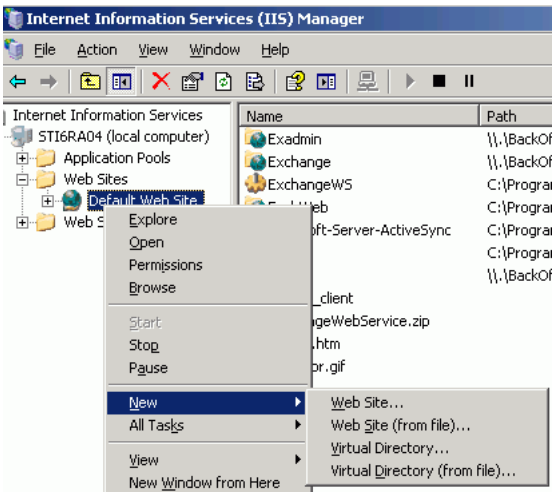
   Run the `RtcApiSetup.msi` installer first, then the side-by-side policy switcher installer (`RtcSxSPolicies.msi`), and restart the system.

2. To install the Oracle RTC Web service for Microsoft Live Communications Server 2005, extract the `owc_lcs.zip` file from the Oracle Fusion Middleware companion CD. It is located in the directory `/Disk1/WebCenter/services/imp/NT`. The zip file contains the following:

   ```
   /Bin

   /images

   ApplicationConfigurationService.asmx

   BlafPlus.css

   ExtAppLogin.aspx

   ExtAppLogin.aspx.cs

   Global.asax

   Log4Net.config

   RTCService.asmx

   Web.Config

   WebcenterTemplate.master
   ```

3. Open the Internet Information Services (IIS) Manager.
4. Expand the server node and then **Web Sites** in the IIS Manager window.
5. Right-click **Default Web Site**, select **New**, and then select **Virtual Directory** to create a site for the Oracle RTC Web service, as shown in Figure 14–1.

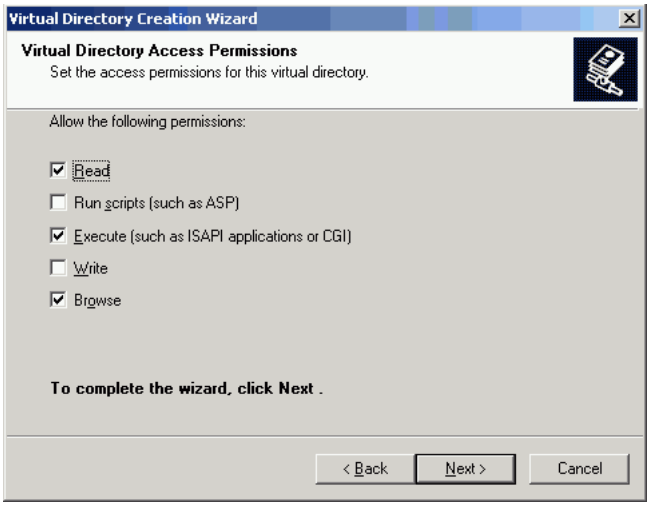
Figure 14–1 Creating a Virtual Directory



The Virtual Directory Creation Wizard displays.

6. Click **Next**.
7. Enter an alias for the virtual directory in the **Alias** field, for example **RTC**.
8. Enter the path to the directory where you extracted the `owc_lcs.zip` file. Alternatively, use the **Browse** button to navigate to that directory.
9. Click **Next**.
10. Ensure that the virtual directory has the Read, Execute, and Browse privileges. (Figure 14–2)

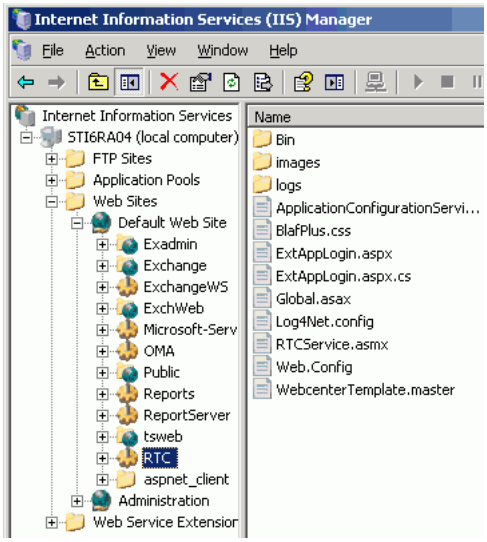
Figure 14–2 Virtual Directory Properties



11. Click **Next**.
12. Click **Finish**.

The newly created virtual directory appears under **Default Web Site** in the Internet Information Services (IIS) Manager window (Figure 14–3).

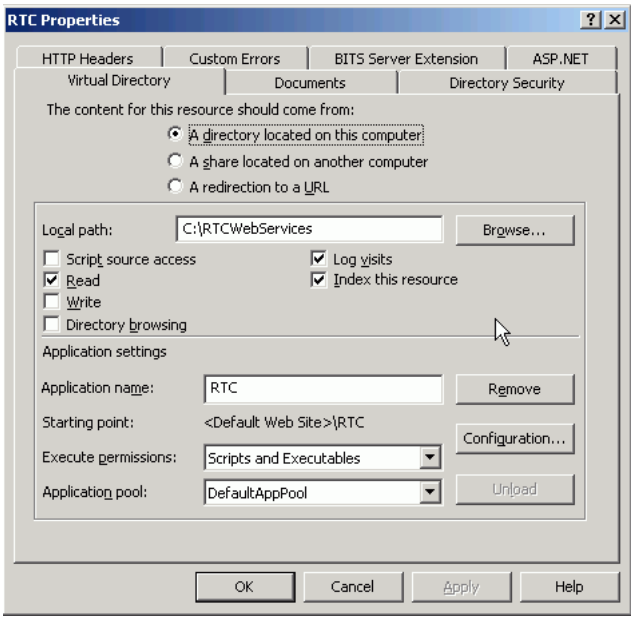
Figure 14–3 Adding a Virtual Directory



13. Right-click the newly created virtual directory for the Oracle RTC Web service, and then select **Properties** to open the Properties dialog.
14. In the Virtual Directory tab, under **Application settings,** click **Create**.

    Notice that the button label changes to **Remove**, and the name of your newly created virtual directory appears in the **Application name** field.
15. Select **Scripts and Executables** from the **Execute permissions** drop-down list (Figure 14–4).

Figure 14–4 Virtual Directory Properties



16. Under the **ASP.NET** tab, select the ASP.NET version as 2.0 or higher from the **ASP.NET version** drop-down list.

    Configure IIS to consume ASP.NET 2.0 applications.

17. Click **OK**.
18. Ensure that the LSC pool name in the LCS connection has been set.
19. Test the Web service by accessing the website using the following URL format:

    `http://localhost/default_website/ApplicationConfigurationService.asmx`

    Where *default_website* refers to the virtual directory that you created for the Oracle RTC Web service.

    For example:

    `http://localhost/RTC/ApplicationConfigurationService.asmx`

14.2.1.3 Microsoft LCS - Security Considerations

You must configure an external application for Microsoft Live Communications Server connections so that users can supply credentials to authenticate themselves on the LCS server.

With a secured application, users get presence status. With LCS, if security is required, then LCS should be on a private trusted network.

LCS provides an option for changing external credentials, which works as an alternative to using an external application. A logged-in user can click any Presence tag and select **Change Credentials** from the menu.

For more information, see Section 14.3.1, "Registering Instant Messaging and Presence Servers Using Fusion Middleware Control."

14.2.2 Microsoft Office Communications Server (OCS) Prerequisites

This section describes the Microsoft Office Communications Server 2007 (OCS) prerequisites as the presence server for instant messaging and presence.

This section includes the following subsections:

14.2.2.1 Microsoft OCS - Installation

Refer to the Microsoft Office Communications Server 2007 documentation for installation information.

14.2.2.2 Microsoft OCS - Configuration

This section includes the following subsections:

- Section 14.2.2.2.7, "Installing WebCenter Portal's Proxy Application"

To use Microsoft OCS 2007 as the presence server for IMP, you must deploy WebCenter Portal's Proxy application for Microsoft OCS 2007 in one of two topologies:

- Simple Deployment – All components reside on the same box
- Remote Deployment – The proxy application and Microsoft OCS reside on separate boxes

14.2.2.2.1 Simple Deployment In this topology, WebCenter Portal's Proxy application is deployed in the Internet Information Services (IIS) server hosted on the OCS box.

1. Install Microsoft Unified Communications Managed API (UCMA) 2.0 on the OCS box.

   For detailed information, see Section 14.2.2.2.6, "Installing UCMA v2.0."

2. Deploy WebCenter Portal's Proxy application on the IIS server.

   This proxy application provides web services for interacting with the OCS server, and for sending and receiving information. WebCenter Portal talks to these web services and presents the data.

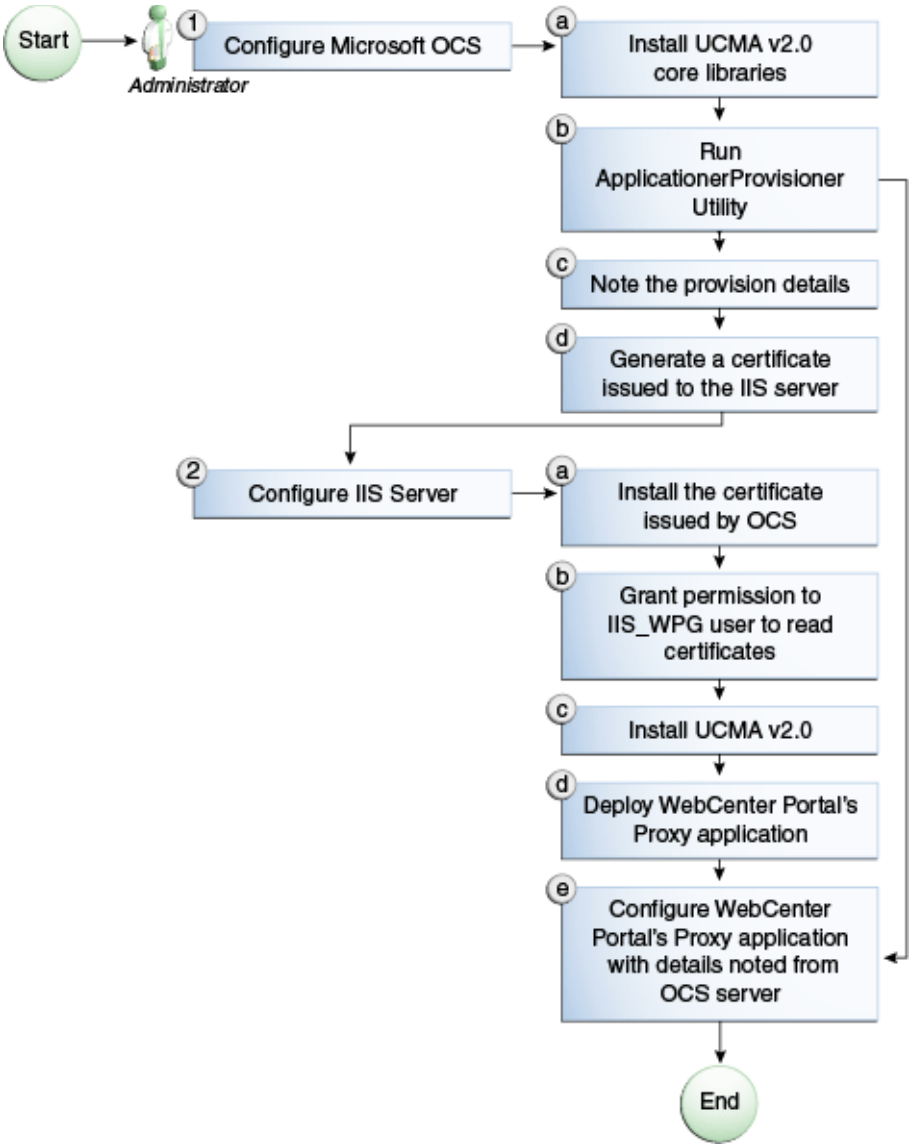
   For detailed information, see Section 14.2.2.2.7, "Installing WebCenter Portal's Proxy Application."

14.2.2.2.2 Remote Deployment In this topology, WebCenter Portal's Proxy application is deployed on an IIS server remote to the OCS box. That is, the IIS server and the OCS server are hosted on separate machines.

Because this proxy application is hosted on a remote box, you must set up a trust between the application and the OCS server. This is known as *provisioning* an application. Provisioning is done through the Application Provisioner utility shipped with Microsoft UCMA v2.0. For more details, see `http://msdn.microsoft.com/en-us/library/dd253360%28office.13%29.aspx`.

Figure 14–5 provides an overview of the steps (including installing UCMA v2.0) to be performed on different deployment entities.

Figure 14–5 Microsoft OCS Configuration - Remote Deployment



The details of these steps are described in the following sections.

14.2.2.2.3 Building Application Provisioner This section lists the steps Microsoft provides for provisioning other IIS servers to access OCS.

1. Install Visual Studio 2008 on any developer box (not necessarily IIS/OCS).
2. Install UCMA version 2.0 on the same box following the steps in Section 14.2.2.2.6, "Installing UCMA v2.0."

   The Application Provisioner application comes with the UCMA SDK.
3. Go to the directory `Sample Applications\Collaboration\ApplicationProvisioner` under the location where you installed UCMA Core (for example, `C:\Program Files\Microsoft Office Communications Server 2007 R2\UCMA SDK 2.0\UCMACore\Sample Applications\Collaboration\ApplicationProvisioner`).

The directory contains the Application Provisioner application.

4. Build the application using Visual Studio 2008.

   This generates the `ApplicationProvisioner.exe` file.

5. Copy the executable file to the OCS box.

14.2.2.2.4 Provisioning WebCenter Portal's Proxy Application on OCS Server

1. Install UCMA v2.0 core libraries on the OCS box.

   Follow the same steps in Section 14.2.2.2.6, "Installing UCMA v2.0," except that after installing Visual C++ 2008 Redistributable, run `OCSCore.msi`.

   This installs the WMI classes required to provision an application.

2. Run the `ApplicationProvisioner.exe` file, generated in the previous section.

   The Application Provisioner dialog appears.

3. In the Application Provisioner dialog, enter `WebCenterProxyApplication` as the name of your application for the Application name, and then click **Find or Create**.

4. In the Create Application Pool dialog, select the Office Communications Server pool for your application in the OCS Pool Fqdn list.

   - For Listening port, enter the listening port for your application (for example, 6001).
   - For Application server Fqdn, enter the fully qualified domain name (FQDN) of the computer on which the application is deployed. (This is the IIS box.)
   - If the application is deployed on two or more computers, then select the Load balanced application check box, and for Load balancer Fqdn, enter the FQDN of the load balancer.

   The application pool now appears in the Application Provisioner dialog.

5. Double-click the server entry.

   The View Server dialog appears. Note the information shown there; that is, Server FQDN, port, and GRUU.

6. Create a certificate on the OCS server with the subject name as the Server FQDN, noted in the previous step, using the Office Communications Server Certificate Wizard.

   This certificate is used to authorize the requests coming from the IIS server.

7. After the certificate is created, view the certificate.

8. On the Details tab click **Copy to File**.

   The Certificate Export Wizard appears.

9. Export the certificate with the private key to a file.

   A `.pfx` (Personal Information Exchange) file with the certificate name is created.

14.2.2.2.5 IIS Server Configuration Because the IIS server hosts WebCenter Portal's Proxy application in the remote deployment scenario, use the information from the previous section to make it a trusted authority.

1. Install the certificate issued by the OCS server with the private key: Copy the .pfx file generated in step 7 under section "Provisioning WebCenter Portal's Proxy

Application on OCS Server" to the IIS box, and double-click it.

The Certificate Import wizard appears.

2. Import the certificate in Personal Folder under LOCAL_MACHINE.
3. Give permission to IIS_WPG user for reading the certificate.

   This is required so that the IIS server has appropriate read access on the certificate. To do this, you can use a utility provided by Microsoft called Windows HTTP Services Certificate Configuration Tool (`http://www.microsoft.com/downloads/details.aspx?familyid=c42e27ac-3409-40e9-8667-c748e422833f&displaylang=en`). Download the utility and install it. This creates an executable called `winhttpcertcfg.exe`. Go to the install location and run the following command to grant permission:

   ```
   winhttpcertcfg.exe -g -c LOCAL_MACHINE\MY -s "<certificate-name>" -a "IIS_WPG"
   ```

4. Make an entry in `C:/WINDOWS/system32/drivers/etc/hosts` for the pool name of the OCS server as follows:

   ```
   <ip-address-of-ocs-box> <poolname-of-ocs-box>
   ```

   For example:

   ```
   10.177.252.146 pool01.example.com
   ```

5. Because the IIS server hosts WebCenter Portal's Proxy application, install Microsoft UCMA v2.0 on it.

   For detailed information, see Section 14.2.2.2.6, "Installing UCMA v2.0."

6. After UCMA is installed, deploy the proxy application on the IIS server.

   WebCenter Portal's Proxy application provides web services for interacting with OCS server, and for sending and receiving information. WebCenter Portal talks to these web services and presents the data.

   For detailed information, see Section 14.2.2.2.7, "Installing WebCenter Portal's Proxy Application."

7. Go to the location where WebCenter Portal's Proxy application was extracted, and open `Web.config` and edit the `appSettings` XML node to add the values noted in Step 7 in previous section.

   Ensure to set value for `RemoteDeployment` to true.

   For example, the `appsettings` XML node should look somewhat like this.

   ```
   <appSettings>
     <add key="ApplicationName" value="WebCenterProxyApplication"/>
     <add key="RemoteDeployment" value="true"/>
     <add key="ApplicationFQDN" value="iis.server.com"/>
     <add key="ApplicationGRUU"
   value="sip:iis.server.com@EXAMPLE.COM;gruu;opaque=srvr:WebCenterProxyApplicatio
   n:7mhSo94PlUK-5Q2bKPLyMAAA"/>
     <add key="ApplicationPort" value="6001"/>
   </appSettings>
   ```

   The trust is established, and WebCenter Portal's Proxy application can talk to OCS.

14.2.2.2.6 Installing UCMA v2.0 Microsoft Unified Communications Managed API v2.0 (UCMA) is an endpoint API that allows advanced developers to build server applications that can interact with the OCS environment.

In a simple deployment, the UCMA is installed on the same box as OCS. In a remote deployment, the OCS core libraries are installed on the OCS box, and the UCMA is installed on the IIS (proxy) box.

1. Download UCMA v2.0 from the following location:
   - For OCS2007 R2 installation (64 bit): http://www.microsoft.com/downloads/details.aspx?FamilyID=b20967b1-6cf5-4a4b-b7ae-622653ac929f&displaylang=en

     Download and run the `UcmaSDKWebDownload.msi` file. This extracts files to the folder `C:\Microsoft Unified Communications Managed API 2.0 SDK Installer package\amd64`.
2. Go to the directory (where the files from the previous step were extracted) and run `vcredist_x86.exe`.

   This step installs run-time components of Visual C++ Libraries required for UCMA APIs.
3. Go to directory called `Setup` and run `UcmaRedist.msi`.

   This step installs the UCMA 2.0 assemblies in the GAC.

14.2.2.2.7 Installing WebCenter Portal's Proxy Application

1. Extract `owc_ocs2007.zip` from the companion CD.

   A directory named `OCSWebServices` is created.
2. Open the Internet Information Services (IIS) Manager.
3. Expand the server node and then **Web Sites** in the Internet Information Services (IIS) Manager.
4. Right-click **Default Web Site**, select **New**, and then select **Virtual Directory** to create a site for the Oracle RTC Web service.

   The Virtual Directory Creation Wizard displays.
5. Click **Next**.
6. Enter an alias for the virtual directory in the **Alias** field, for example `RTC`.
7. Enter the path to the directory extracted from `owc_ocs2007.zip` file.

   If you had extracted the zip file in C:\, then the path supplied should be C:\OCSWebServices. Alternatively, use the Browse button to navigate to that directory.
8. Click **Next**.
9. Ensure that the virtual directory has the Read, Execute, and Browse privileges.
10. Click **Next**.
11. Click **Finish**.

    The newly created virtual directory appears under Default Web Site in the Internet Information Services (IIS) Manager window.
12. Right-click the newly created virtual directory for the Oracle RTC Web service, and then select **Properties** to open the Properties dialog.
13. In the Virtual Directory tab, under Application settings, click **Create**.

    Notice that the button label changes to Remove, and the name of your newly created virtual directory appears in the Application name field.

14. Select **Scripts and Executables** from the Execute permissions drop-down list.
15. Under the ASP.NET tab, select the ASP.NET version as 2.0 or higher from the ASP.NET version drop-down list.

    IIS should be configured to consume ASP.NET 2.0 applications.
16. Click **OK**.
17. Test the Web service by accessing the Web site using the following URL format: `http://localhost/default_website/OCSWebService.asmx`.

    where *`default_website`* is the virtual directory you created for the Oracle RTC Web service

    For example:

    ```
    http://localhost/RTC/OCSWebService.asmx
    ```

14.2.2.3 Microsoft OCS - Security Considerations

You must configure an external application for Microsoft Office Communications Server connections so that users can supply credentials to authenticate themselves on the OCS server.

With a secured application, users get presence status. With OCS, if security is required, then OCS should be on a private trusted network.

OCS provides an option for changing external credentials, which works as an alternative to using an external application. A logged-in user can click any Presence tag and select **Change Credentials** from the menu.

For more information, see Section 14.3.1, "Registering Instant Messaging and Presence Servers Using Fusion Middleware Control."

14.2.3 Microsoft Lync Prerequisites

This section describes the Microsoft Lync 2010 prerequisites as the presence server for instant messaging and presence.

This section includes the following subsections:

- Section 14.2.3.1, "Microsoft Lync - Installation"
- Section 14.2.3.2, "Microsoft Lync - Configuration"
- Section 14.2.3.3, "Microsoft Lync - Security Considerations"

14.2.3.1 Microsoft Lync - Installation

Refer to the Microsoft Lync 2010 documentation for installation information.

14.2.3.2 Microsoft Lync - Configuration

This section includes the following subsections:

- Section 14.2.3.2.1, "Simple Deployment"
- Section 14.2.3.2.2, "Remote Deployment"
- Section 14.2.3.2.3, "Building Application Provisioner"
- Section 14.2.3.2.4, "Provisioning WebCenter Portal's Proxy Application on Lync Server"
- Section 14.2.3.2.5, "Adding AllowedDomains Using WBemTest"

- Section 14.2.3.2.6, "Migrating Trusted Service Entries Using Topology Builder or PowerShell Cmdlets"
- Section 14.2.3.2.7, "IIS Server Configuration"
- Section 14.2.3.2.8, "Installing UCMA v2.0"
- Section 14.2.3.2.9, "Installing WebCenter Portal's Proxy Application"

Configuration for Microsoft Lync is similar to configuration for Microsoft OCS.

To use Microsoft Lync 2010 as the presence server for IMP, you must deploy WebCenter Portal's Proxy application for Microsoft Lync 2010 in one of two topologies:

- Simple Deployment – All components reside on the same box
- Remote Deployment – The proxy application and Microsoft Lync reside on separate boxes

14.2.3.2.1 Simple Deployment In this topology, WebCenter Portal's Proxy application is deployed in the Internet Information Services (IIS) server hosted on the Lync box.

1. Install Microsoft Unified Communications Managed API (UCMA) 2.0 on the Lync box.

   For detailed information, see Section 14.2.3.2.8, "Installing UCMA v2.0."

2. Deploy WebCenter Portal's Proxy application on the IIS server.

   This proxy application provides web services for interacting with the Lync server, and for sending and receiving information. WebCenter Portal talks to these web services and presents the data. For detailed information, see Section 14.2.3.2.9, "Installing WebCenter Portal's Proxy Application."

14.2.3.2.2 Remote Deployment In this topology, WebCenter Portal's Proxy application is deployed on an IIS server remote to the Lync box. That is, the IIS server and the Lync server are hosted on separate machines.

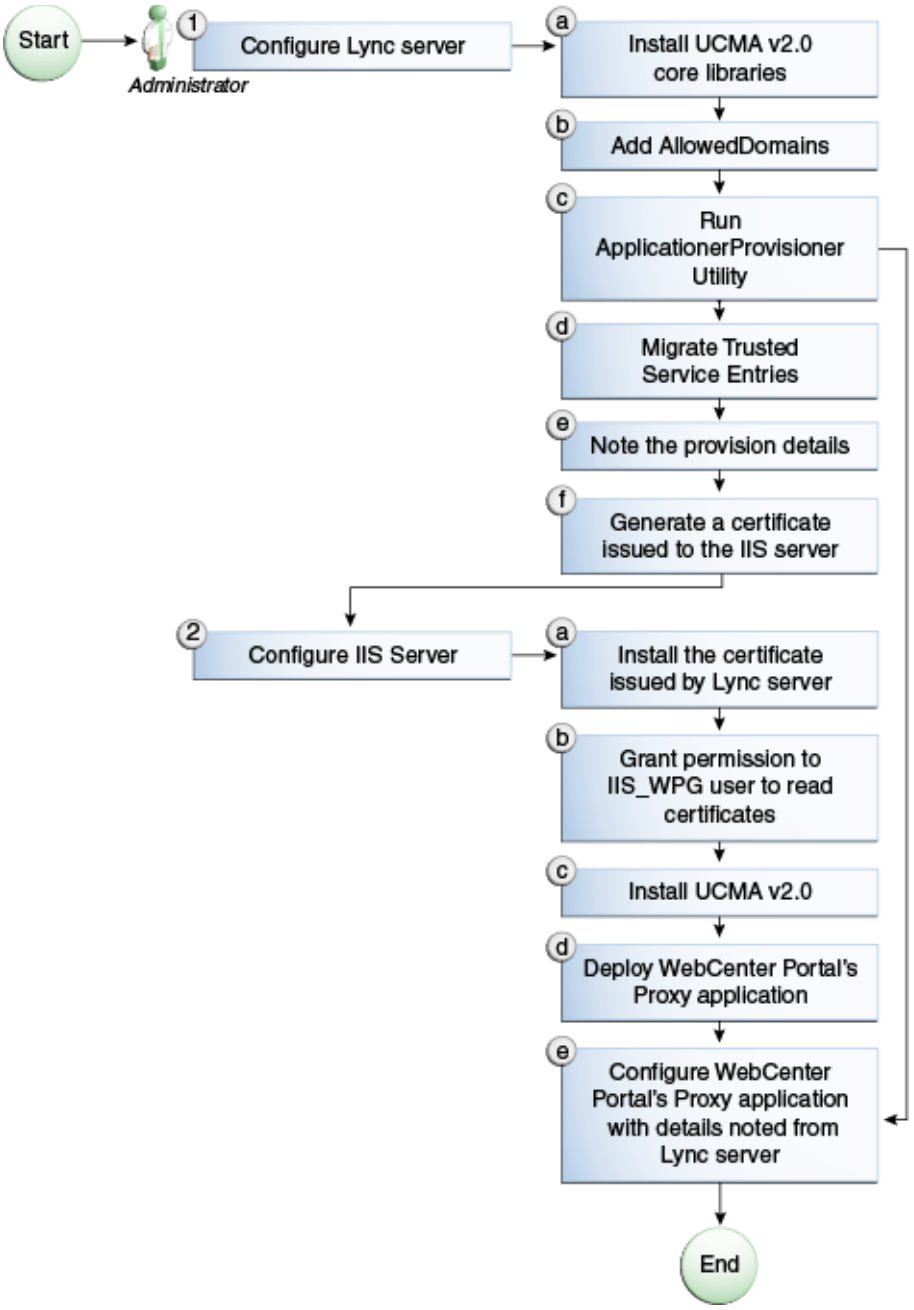
Because this proxy application is hosted on a remote box, you must set up a trust between the application and the Lync server. This is known as *provisioning* an application. Provisioning is done through the Application Provisioner utility shipped with Microsoft UCMA v2.0.

See Also:

`http://msdn.microsoft.com/en-us/library/dd253360%28office.13%29.aspx`

Figure 14–6 provides an overview of the steps (including installing UCMA v2.0) to be performed on different deployment entities.

Figure 14–6 Microsoft Lync Configuration - Remote Deployment



The details of these steps are described in the following sections.

14.2.3.2.3 Building Application Provisioner This section lists the steps Microsoft provides for provisioning other IIS servers to access Lync.

1. Install Visual Studio 2008 on any developer box (not necessarily IIS/Lync).
2. Install UCMA version 2.0 on the same box following the steps in Section 14.2.3.2.8, "Installing UCMA v2.0."

   The Application Provisioner application comes with the UCMA SDK.
3. Go to the directory `Sample Applications\Collaboration\ApplicationProvisioner` under the

location where you installed UCMA Core (for example, `C:\Program Files\Microsoft Lync 2010 R2\UCMA SDK 2.0\UCMACore\Sample Applications\Collaboration\ApplicationProvisioner`).

4. Open the application in Visual Studio 2008 and edit the `Application.cs` file as per `http://msdn.microsoft.com/en-us/library/gg448038.aspx`.
5. Build the application using Visual Studio 2008.

   This generates the `ApplicationProvisioner.exe` file.
6. Copy the executable file to the Lync box.

14.2.3.2.4 Provisioning WebCenter Portal's Proxy Application on Lync Server

1. Run the `OCSWMIBC.msi` file that comes with the Lync setup package.

   When a UCMA 2.0 application is deployed directly against Lync Server 2010, the SIP domains used in the Lync Server 2010 environment must be added to the Office Communications Server 2007 R2 SIP domain list *before* you run the `Merge-CsLegacyTopology` cmdlet. The application is deployed as if it were being deployed against OCS 2007 R2, then migrated to run against Lync Server 2010. To add the domains, see Section 14.2.3.2.5, "Adding AllowedDomains Using WBemTest."
2. Run the `ApplicationProvisioner.exe` file, generated in the previous section.

   The Application Provisioner dialog appears.
3. In the Application Provisioner dialog, enter `WebCenterProxyApplication` as the name of your application for the Application name, and then click **Find or Create**.
4. In the Create Application Pool dialog, select the pool for your application in the Lync Pool Fqdn list.
   - For Listening port, enter the listening port for your application (for example, 6001).
   - For Application server Fqdn, enter the fully qualified domain name (FQDN) of the computer on which the application is deployed. (This is the IIS box.)
   - If the application is deployed on two or more computers, then select the Load balanced application checkbox, and for Load balancer Fqdn, enter the FQDN of the load balancer.

   The application pool now appears in the Application Provisioner dialog.
5. Double-click the server entry.

   The View Server dialog appears. Note the information shown there; that is, Server FQDN, port, and GRUU.
6. Migrate the newly-created trusted entry to Lync Server 2010.

   See Section 14.2.3.2.6, "Migrating Trusted Service Entries Using Topology Builder or PowerShell Cmdlets."
7. Create a certificate on the Lync server with the subject name as the Server FQDN noted in the previous step using the Lync Certificate Wizard.

   This certificate is used to authorize the requests coming from the IIS server.
8. After the certificate is created, view the certificate.
9. On the Details tab click **Copy to File**.

The Certificate Export Wizard appears.

10. Export the certificate with the private key to a file.

    A `.pfx` (Personal Information Exchange) file with the certificate name is created.

14.2.3.2.5 Adding AllowedDomains Using WBemTest

1. To start `WBemTest.exe`, type `WBemTest` in a command prompt window and press **Enter**.
2. In the Windows Management Instrumentation Tester dialog, click **Connect**.
3. In the Connect dialog, click **Connect**.
4. In the Windows Management Instrumentation Tester dialog, click **Enum Classes**.
5. In the Superclass Info dialog, click **OK**.
6. In the Query Result dialog, scroll down to **MSFT_SIPDomainData()**, and double-click this entry.
7. In the Object editor for MSFT_SIPDomainData dialog, click **Instances**.

   The Query Result dialog appears, displaying the InstanceIDs for any instances of the MSFT_SIPDomainData WMI class. These entries are the AllowedDomain entries.
8. To add AllowedDomain entries, click **Add**.
9. In the Instance of MSFT_SIPDomainData dialog, in the Properties listbox, double-click **Address**.
10. In the Property Editor dialog, select the **Not NULL** radio button.
11. In the Value text input pane, enter the Lync server domain; for example, `contoso.com`, and click **Save Property**.
12. In the Instance of MSFT_SIPDomainData dialog, in the Properties listbox, double-click **Authoritative**, make sure that the Authoritative property is not Null and is set to `False`, and then click **Save Property**.
13. In the Instance of MSFT_SIPDomainData dialog, in the Properties listbox, double-click **Default Domain**, make sure that the Default Domain property is not Null and is set to `True`, then click **Save Property**.
14. In the Instance of MSFT_SIPDomainData dialog, click **Save Object**.

14.2.3.2.6 Migrating Trusted Service Entries Using Topology Builder or PowerShell Cmdlets

To migrate trusted service entries using Microsoft Lync Server 2010 Topology Builder:

1. Launch Microsoft Lync Server 2010, Topology Builder.
2. After the existing topology is loaded, under Action, select Merge 2007 or 2007 R2 Topology.
3. Go through the resulting wizard, keeping the default options.
4. Select Publish Topology and complete the wizard, as in the previous step.
5. After the wizard has finished, check that it completed successfully.

   There should be no errors in the user interface.

To migrate trusted service entries using Microsoft Lync Server 2010 PowerShell Cmdlets:

1. From the Start menu, in the Microsoft Lync Server 2010 program group, open Lync Server Management Shell.
2. Run the following PowerShell cmdlet:

   ```
   Merge-CsLegacyTopology -TopologyXmlFileName D:\output.xml
   ```

3. Run the following PowerShell cmdlet:

   ```
   Publish-CsTopology -FileName D:\output.xml
   ```

14.2.3.2.7 IIS Server Configuration Because the IIS server hosts WebCenter Portal's Proxy application in the remote deployment scenario, use the information from the previous section to make it a trusted authority.

1. Install the certificate issued by the Lync server with the private key: Copy the .pfx file generated in step 7 under section "Provisioning WebCenter Portal's Proxy Application on Lync Server" to the IIS box, and double-click it.

   The Certificate Import wizard appears.
2. Import the certificate in Personal Folder under LOCAL_MACHINE
3. Make an entry in `C:/WINDOWS/system32/drivers/etc/hosts` for the pool name of the Lync server as follows:

   <ip-address-of-lync-box> <poolname-of-lync-box>

   For example:

   ```
   10.177.252.146 pool01.example.com
   ```

4. Because the IIS server hosts WebCenter Portal's Proxy application, install Microsoft UCMA v2.0 on it.

   For detailed information, see Section 14.2.3.2.8, "Installing UCMA v2.0."
5. After UCMA is installed, deploy this proxy application on the IIS server.

   WebCenter Portal's Proxy application provides web services for interacting with Lync, and for sending and receiving information. WebCenter Portal talks to these web services and presents the data. For detailed information, see Section 14.2.3.2.9, "Installing WebCenter Portal's Proxy Application."
6. Go to the location where WebCenter Portal's Proxy application was extracted, and open `Web.config` and edit the `appSettings` XML node to add the values noted in Step 7 in the previous section (Section 14.2.2.2.5, "IIS Server Configuration").

   Make sure to set the value for `RemoteDeployment` to true. For example, the `appsettings` XML node should look somewhat like this.

   ```
   <appSettings>
     <add key="ApplicationName" value="WebCenterProxyApplication"/>
     <add key="RemoteDeployment" value="true"/>
     <add key="ApplicationFQDN" value="iis.server.com"/>
     <add key="ApplicationGRUU"
   value="sip:iis.server.com@EXAMPLE.COM;gruu;opaque=srvr:WebCenterProxyApplicatio
   n:7mhSo94PlUK-5Q2bKPLyMAAA"/>
     <add key="ApplicationPort" value="6001"/>
   </appSettings>
   ```

Note: If you see the following exception in the log file:

```
ErrorCode = -2146893039
FailureReason = NoAuthenticatingAuthority
e.Message = "Unable to perform authentication of credentials."
base {Microsoft.Rtc.Signaling.FailureResponseException} = {"Unable
to perform authentication of credentials."}
InnerException = {"NegotiateSecurityAssociation failed, error:
\-2146893039"}
```

then add the following entry to `Web.config`:

```
<identity impersonate="true" userName="Administrator"
password="MyPassword*"/>
```

where `username` is the administrator's user name, and `password` is the administrator's password.

The trust is established, and WebCenter Portal's Proxy application can talk to the Lync server.

14.2.3.2.8 Installing UCMA v2.0

Microsoft Unified Communications Managed API v2.0 (UCMA) is an endpoint API that allows advanced developers to build server applications that can interact with the Lync environment.

In a simple deployment, the UCMA is installed on the same box as Lync. In a remote deployment, the Lync core libraries are installed on the Lync box, and the UCMA is installed on the IIS (proxy) box.

1. Download UCMA v2.0 installation from the following location: `http://www.microsoft.com/downloads/details.aspx?FamilyID=b20967b1-6cf5-4a4b-b7ae-622653ac929f&displaylang=en`
2. Download and run the `UcmaSDKWebDownload.msi` file.

   Setup files are extracted to the folder `C:\Microsoft Unified Communications Managed API 2.0 SDK Installer package\amd64`
3. Go to the directory (where the files from the previous step were extracted) and run `vcredist_x86.exe`.

   Run-time components of Visual C++ Libraries, required for UCMA APIs, are installed.
4. Go to the directory called `Setup` and run `UcmaRedist.msi`.

   UCMA 2.0 assemblies in the GAC are installed.

14.2.3.2.9 Installing WebCenter Portal's Proxy Application

1. Extract `owc_ocs2007.zip` from the companion CD.

   A directory named `OCSWebServices` is created.
2. Open the Internet Information Services (IIS) Manager.
3. Expand the server node and then **Sites** in the IIS Manager.
4. Right-click **Lync Internal Web Site**, and then select **Add Application.**
5. In the Add Application wizard, enter an alias for the virtual directory in the **Alias** field, for example `RTC`.

6. Enter the path to the directory extracted from the `owc_ocs2007.zip` file, and then click **OK**.

   For example, if you extracted the zip file in C:\, then enter `C:\OCSWebServices`. Alternatively, use the **Browse** button to navigate to that directory. Click **OK**.

7. Right-click the newly created application and select **Edit Permissions** to open the Properties dialog.

8. In the Security tab, edit permissions to grant user Everyone read permission.

9. Test the Web service by accessing the website using the following URL format: `http://localhost/`*`lync_internal_web_site`*`/OCSWebService.asmx`.

   where *`lync_internal_web_site`* is the virtual directory you created for the Oracle RTC Web service.

   For example:

   ```
   http://localhost/RTC/OCSWebService.asmx
   ```

14.2.3.3 Microsoft Lync - Security Considerations

You must configure an external application for Microsoft Lync connections so that users can supply credentials to authenticate themselves on the Lync server.

With a secured application, users get presence status. With Lync, if security is required, then Lync should be on a private trusted network.

Lync provides an option for changing external credentials, which works as an alternative to using an external application. A logged-in user can click any Presence tag and select **Change Credentials** from the menu.

For more information, see Section 14.3.1, "Registering Instant Messaging and Presence Servers Using Fusion Middleware Control."

14.3 Registering Instant Messaging and Presence Servers

You can register multiple presence server connections with WebCenter Portal, but only one of them is active at a time.

To start using the new (active) presence server you must restart the managed server on which WebCenter Portal is deployed.

This section includes the following subsections:

- Section 14.3.1, "Registering Instant Messaging and Presence Servers Using Fusion Middleware Control"
- Section 14.3.2, "Registering Instant Messaging and Presence Servers Using WLST"

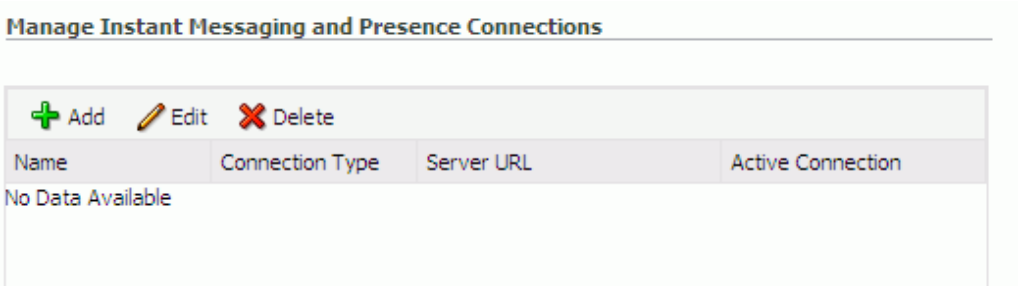
14.3.1 Registering Instant Messaging and Presence Servers Using Fusion Middleware Control

To register a presence server connection:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"

2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework application - From the **Application Deployment** menu, select **WebCenter Portal > Service Configuration**.
3. On the WebCenter Portal Service Configuration page, select **Instant Messaging and Presence**.
4. To connect to a new presence server, click **Add** (Figure 14–7).

Figure 14–7 Configuring Instant Messaging and Presence



5. Enter a unique name for this connection, specify the presence server type, and indicate whether this connection is the active (or default) connection for the application (Table 14–1).

Table 14–1 Instant Messaging and Presence Connection - Name

Field	Description
Connection Name	Enter a unique name for the connection. The name must be unique (across all connection types) within WebCenter Portal.
Connection Type	Specify the type of presence server: ■ Microsoft Live Communications Server (LCS) ■ Microsoft Office Communications Server 2007 (OCS) Out-of-the-box, WebCenter Portal supports Microsoft LCS, OCS, and Lync. **Note**: Microsoft Lync connections use the Microsoft Office Communications Server 2010 connection type. (Oracle Beehive Server connections are not supported in this release.)
Active Connection	Select to use this connection in WebCenter Portal for instant messaging and presence. While you can register multiple presence server connections for an application, only one connection is used by IMP—the default (or active) connection.

6. Enter connection details for the server hosting instant messaging and presence (Table 14–2).

Table 14–2 Instant Messaging and Presence Connection - Connection Details

Field	Description
Server URL	Enter the URL of the server hosting instant messaging and presence. For example: `http://myocshost.com:8888`
User Domain	(OCS/Lync Only) Enter the name of the Active Directory domain (on the Microsoft Office Communications Server) that is associated with this connection. The user domain is mandatory for OCS/Lync connections. Refer to Microsoft documentation for details on the user domain.
Pool Name	Enter the name of the pool that is associated with this connection. The pool name is mandatory. Refer to Microsoft documentation for details on the pool name.
Associated External Application	Associate the instant messaging and presence server with an external application. External application credential information is used to authenticate users against the instant messaging and presence server. An external application is mandatory. You can select an existing external application from the list, or click **Create New** to configure a new external application. The external application you configure for instant messaging and presence must use the `POST` authentication method, and specify an additional field named `Account` (Name property) that is configured to `Display to User` (checked). For more information, see Chapter 23, "Managing External Applications."

7. Enter a timeout in the Advanced Configuration field (Table 14–4).

Table 14–3 Instant Messaging and Presence Connection - Advanced Configuration

Field	Description
Connection Timeout (seconds)	Specify a suitable timeout for the connection. This is the length of time (in seconds) WebCenter Portal waits for a response from the presence server before issuing a connection timeout message. The default is -1 which means that the default is used. The default is 10 seconds.

8. Sometimes, additional parameters are required to connect to the presence server.

   If additional parameters are required to connect to the presence server, expand **Additional Properties** and enter details as required (Table 14–4).

Table 14–4 Instant Messaging and Presence Connection - Additional Properties

Field	Description
Add	Click **Add** to specify an additional connection parameter: ▪ **Property Name** -Enter the name of the connection property. ▪ **Property Value** - Enter the default value for the property. ▪ **Is Property Secured** - Indicate whether encryption is required. When selected, the property value is stored securely using encryption. For example, select this option to secure the `admin.password` property where the value is the actual password.
Delete	Click **Delete** to remove a selected property. Select the correct row before clicking **Delete**. **Note:** Deleted rows appear disabled until you click **OK**.

9. Click **OK** to save this connection.
10. To start using the new (active) connection you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed.

    For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

14.3.2 Registering Instant Messaging and Presence Servers Using WLST

Use the WLST command `createIMPConnection` to create a presence server connection. For command syntax and examples, see the "createIMPConnection" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

To configure instant messaging and presence to actively use a new IMP connection, set `default=true`. For more information, see Section 14.4.2, "Choosing the Active Connection for Instant Messaging and Presence Using WLST."

Note: To start using the new (active) connection you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed. For more information, see the "Starting and Stopping Managed Servers Using the Command Line" section in *Oracle Fusion Middleware Administrator's Guide*.

14.4 Choosing the Active Connection for Instant Messaging and Presence

You can register multiple instant messaging and presence server connections with WebCenter Portal, but only one connection is active at a time. The *active connection* becomes the back-end presence server for WebCenter Portal.

This section includes the following subsections:

- Section 14.4.1, "Choosing the Active Connection for Instant Messaging and Presence Using Fusion Middleware Control"

- Section 14.4.2, "Choosing the Active Connection for Instant Messaging and Presence Using WLST"

14.4.1 Choosing the Active Connection for Instant Messaging and Presence Using Fusion Middleware Control

To change the active connection:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal > Service Configuration**.
3. On the WebCenter Portal Services Configuration page, select **Instant Messaging and Presence**.

   The Manage Instant Messaging and Presence Connections table indicates the current active connection, if any.
4. Select the connection you want to make the active (or default) connection, and then click **Edit**.
5. Select the **Active Connection** check box.
6. Click **OK** to update the connection.
7. To start using the new (active) connection you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed.

   See Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

14.4.2 Choosing the Active Connection for Instant Messaging and Presence Using WLST

Use the WLST command `setIMPConnection` with `default=true` to activate an existing presence server connection. For command syntax and examples, see the "setIMPConnection" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

To disable a presence server connection, either delete it, make another connection the 'active connection,' or use the `removeIMPServiceProperty` command:

```
removeIMPServiceProperty('appName='webcenter', property='selected.connection')
```

Using this command, connection details are retained but the connection is no longer named as an active connection. For more information, see the "removeIMPServiceProperty" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Note: To start using this active connection you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed. For more information, see the "Starting and Stopping Managed Servers Using the Command Line" section in *Oracle Fusion Middleware Administrator's Guide*.

14.5 Modifying Instant Messaging and Presence Connection Details

You can modify instant messaging and presence server connection details at any time.

To start using an updated (active) connection you must restart the managed server on which WebCenter Portal is deployed.

This section includes the following subsections:

- Section 14.5.1, "Modifying Instant Messaging and Presence Connections Details Using Fusion Middleware Control"
- Section 14.5.2, "Modifying Instant Messaging and Presence Connections Details Using WLST"

14.5.1 Modifying Instant Messaging and Presence Connections Details Using Fusion Middleware Control

To update connection details for an instant messaging and presence server:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal > Service Configuration**.
3. On the WebCenter Portal Service Configuration page, select **Instant Messaging and Presence**.
4. Select the connection name, and click **Edit**.
5. Edit connection details, as required.

   For detailed parameter information, see Table 14–2, " Instant Messaging and Presence Connection - Connection Details".
6. Click **OK** to save your changes.
7. To start using the updated (active) connection you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed.

   For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

14.5.2 Modifying Instant Messaging and Presence Connections Details Using WLST

Use the WLST command `setIMPConnection` to edit presence server connection details. For command syntax and examples, see the "setIMPConnection" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

If additional parameters are required to connect to your presence server, then use the `setIMPConnectionProperty` command. For more information, see the "setIMPConnectionProperty" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Note: To start using the updated (active) connection you must restart the managed server on which WebCenter Portal and your Portal Framework application is deployed. For more information, see the "Starting and Stopping Managed Servers Using the Command Line" section in *Oracle Fusion Middleware Administrator's Guide*.

14.6 Deleting Instant Messaging and Presence Connections

You can delete instant messaging and presence connections at any time, but use caution when deleting the active connection. When you delete the active connection, user presence options are not available, as these require a back-end instant messaging and presence server.

When you delete a connection, consider deleting the external application associated with instant messaging and presence *if* the application's sole purpose was to support it. For more information, see Section 23.6, "Deleting External Application Connections."

This section includes the following subsections:

- Section 14.6.1, "Deleting Instant Messaging and Presence Connections Using Fusion Middleware Control"
- Section 14.6.2, "Deleting Instant Messaging and Presence Connections Using WLST"

14.6.1 Deleting Instant Messaging and Presence Connections Using Fusion Middleware Control

To delete an instant messaging and presence server connection:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal > Service Configuration**.

3. On the WebCenter Portal Service Configuration page, select **Instant Messaging and Presence**.
4. Select the connection name, and click **Delete**.
5. To make this change you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed.

   For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

> **Note:** Before restarting the managed server, mark another connection as active; otherwise, Instant Messaging and Presence is disabled.

14.6.2 Deleting Instant Messaging and Presence Connections Using WLST

Use the WLST command `deleteConnection` to remove a presence server connection. For command syntax and examples, see the "deleteConnection" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

14.7 Setting Up Instant Messaging and Presence Defaults

Use the WLST command `setIMPServiceProperty` to set defaults for IMP:

- `selected.connection`: Connection used by instant messaging and presence.
- `rtc.cache.time`: Cache timeout for instant messaging and presence data.
- `resolve.display.name.from.user.profile`: Determines what to display if user display names are missing. When set to `0`, and display name information is unavailable, only the user name displays in the application. When set to `1`, and display name information is unavailable, display names are read from user profile data. Setting this option to `1` impacts performance. The default setting is `0`.

  Display names are not mandatory in presence data. If the application does not always provide display names by default and you consider this information important, set `resolve.display.name.from.user.profile` to `1` so that display names always display.
- `im.address.resolver.class`: Resolver implementation used to map user names to IM addresses and IM addresses to user names. The default setting is `oracle.webcenter.collab.rtc.IMPAddressResolverImpl`. This implementation looks for IM addresses in the following places and order:
  - User Preferences
  - User Credentials
  - User Profiles
- `im.address.profile.attribute`: User profile attribute used to determine a user's IM address. The default setting is `BUSINESS_EMAIL`. Users can change this default with `im.address.profile.attribute`.

For command syntax and detailed examples, see the "setIMPServiceProperty" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

14.8 Testing Instant Messaging and Presence Connections

Oracle RTC web services expose a set of web methods that you can invoke to test validity. To verify a connection, try accessing the web service endpoints. The following examples assume the application context path is `/RTC`:

- *protocol*://*host*/RTC/ApplicationConfigurationService.asmx
- *protocol*://*host*/RTC/RTCService.asmx
- *protocol://host*/RTC/OCSWebService.asmx

15

Managing Mail

This chapter describes how to configure and manage mail for WebCenter Portal. It also describes how to configure the "Send Mail" feature, which allows application assets to send mail directly from them. The Send Mail feature does not require mail. That is, even if the Mail component has not been configured in WebCenter Portal, users can send mail notifications. For more information about using Send Mail notifications, see the "About the Send Mail Feature" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

Always use Fusion Middleware Control or WLST command-line tool to review and configure back-end servers for WebCenter Portal and Portal Framework applications. Any changes that you make to these applications, postdeployment, are stored in MDS metatdata store as customizations. See Section 1.3.5, "Oracle WebCenter Portal Configuration Considerations."

> **Note:** Configuration changes for mail, through Fusion Middleware Control or using WLST, are not dynamic, so you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed for your changes to take effect. See Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

This chapter includes the following topics:

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console and the `Administrator` role in the deployed application:

- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.
- **Portal Framework application**: `Administrator` role granted through the Administration Console.

For more information about roles and permissions, see Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

15.1 About Mail Server Connections

Oracle WebCenter Portal supports the Microsoft Exchange Server or any mail server that supports IMAP4 and SMTP. To enable users to access mail and perform basic operations such as read, reply, and forward within WebCenter Portal or your own Portal Framework application, you must first register the appropriate mail server. Mail is not configured out-of-the-box.

You can register multiple mail server connections:

- **WebCenter Portal** supports multiple mail connections. The mail connection marked *active* is the default connection for mail in WebCenter Portal. All additional connections are offered as alternatives; users can choose which one they want to use through user preferences.
- **Portal Framework applications** use only one mail connection—the connection marked *active*. Any additional connections are ignored.

15.2 Configuration Roadmaps for Mail

Use the roadmaps in this section as an administrator's guide through the configuration process:

- **Roadmap - Configuring Mail for WebCenter Portal**

  Figure 15–1 and Table 15–1 provide an overview of the prerequisites and tasks required for mail to work in WebCenter Portal.

Figure 15–1 Configuring Mail

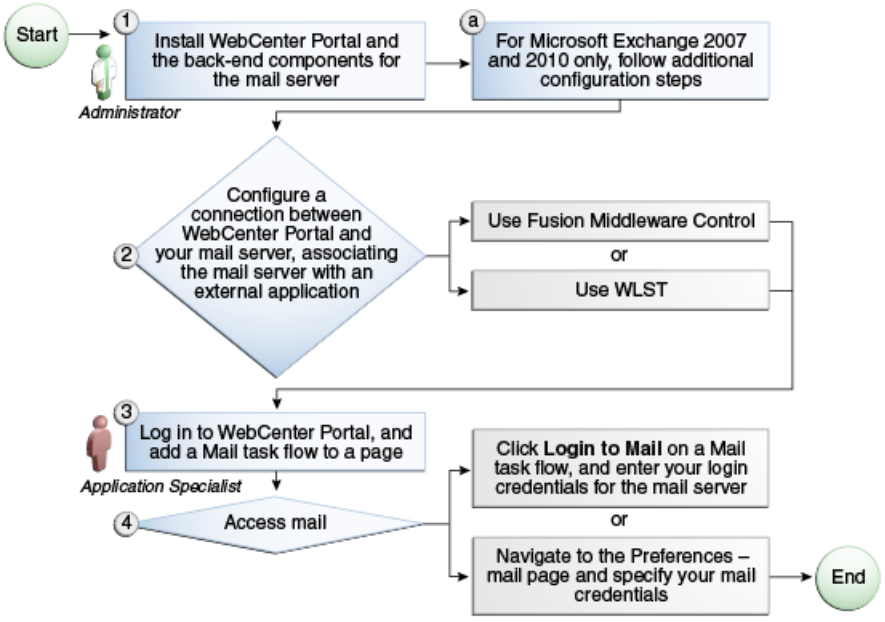


Table 15–1 Configuring Mail for WebCenter Portal

Actor	Task	Sub-task
Administrator	1. Install WebCenter Portal and the back-end components for the mail server (see Mail Server - Installation)	1.a For Microsoft Exchange 2007 only, follow additional configuration steps (see Configuring Microsoft Exchange Server 2007 for WebCenter Portal)
	2. Configure a connection between WebCenter Portal and your mail server -- associating the mail server with an external application -- using one of the following tools: ▪ Fusion Middleware Control (see Registering Mail Servers Using Fusion Middleware Control) ▪ WLST (see Registering Mail Servers Using WLST)	
End User	3. Add the Mail task flow to a page.	
	4. Access mail with one of the following methods: ▪ Click **Login to Mail** on a Mail task flow, and entering your login credentials for the mail server ▪ Navigate to the Preferences - Mail page and specify your mail credentials	

- **Roadmap - Configuring Mail for Portal Framework Applications**

  Figure 15–2 and Table 15–2 provide an overview of the prerequisites and tasks required for mail to work in Portal Framework applications.

Figure 15–2 Configuring Mail for Portal Framework Applications

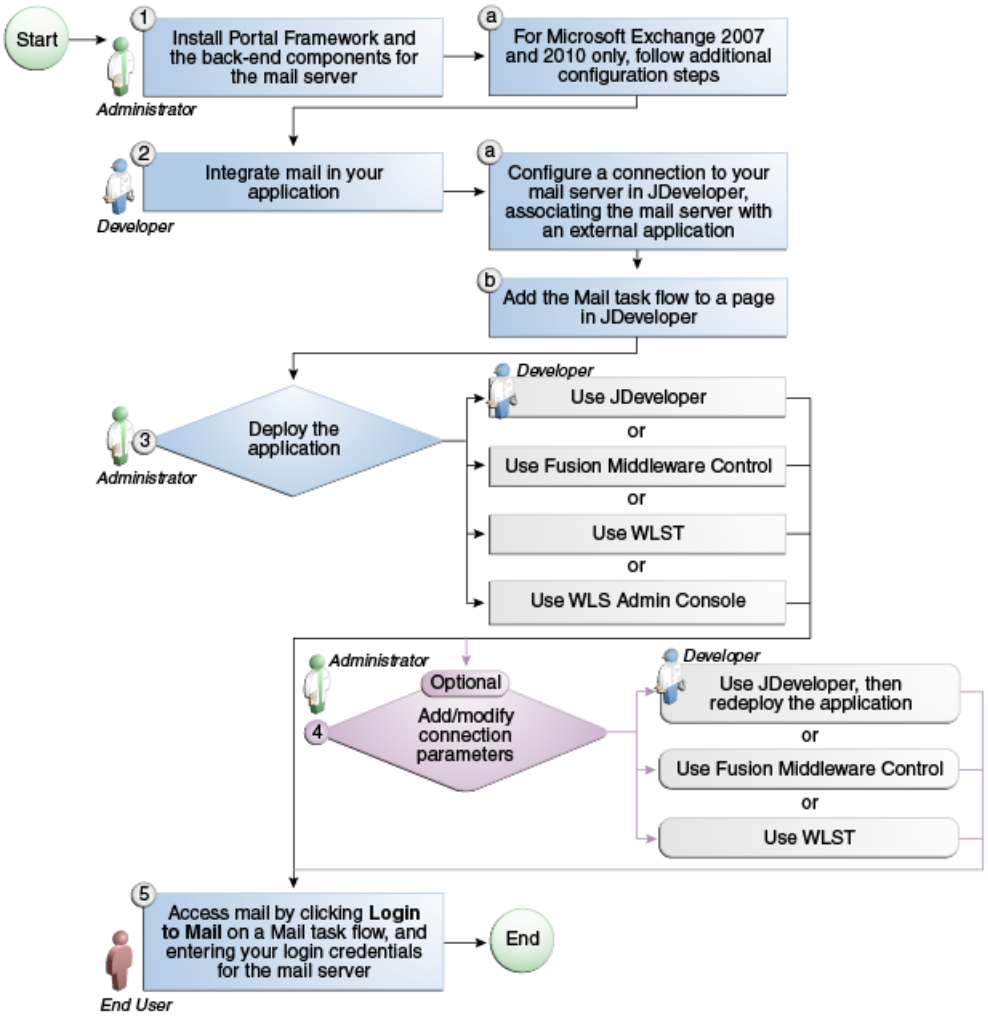


Table 15–2 Configuring Mail for Framework Applications

Actor	Task	Sub-task
Administrator	1. Install WebCenter Portal and the back-end components for the mail server (see Mail Server - Installation)	1.a For Microsoft Exchange 2007 only, follow additional configuration steps (see Configuring Microsoft Exchange Server 2007 for WebCenter Portal)
Developer	2. Integrate mail in your Framework application.	2.a Configure a connection to your mail server in JDeveloper, associating the mail server with an external application
		2.b Add the Mail task flow to a page in JDeveloper

Table 15–2 (Cont.) Configuring Mail for Framework Applications

Actor	Task	Sub-task
Developer or Administrator	3. Deploy the Framework application using one of the following tools: ■ JDeveloper (Developer) ■ Fusion Middleware Control (Administrator) ■ WLST (Administrator) ■ WLS Admin Console (Administrator)	
Developer or Administrator	4. (Optional) Add/modify connection parameters using one of the following tools: ■ JDeveloper, then redeploy the application (Developer) ■ Fusion Middleware Control (Administrator) ■ WLST (Administrator)	
End User	5. Access mail by clicking **Login to Mail** on a Mail task flow, and entering your login credentials for the mail server	

15.3 Mail Server Prerequisites

This section includes the following subsections:

- Section 15.3.1, "Mail Server - Installation"
- Section 15.3.2, "Mail Server - Configuration"
- Section 15.3.3, "Mail Server - Security Considerations"
- Section 15.3.4, "Mail Server - Limitations"

15.3.1 Mail Server - Installation

See your mail server documentation for installation information.

15.3.2 Mail Server - Configuration

For WebCenter Portal, you can allow WebCenter Portal to create and manage portal distribution lists. This feature is supported only with Microsoft Exchange.

If enabled, a portal distribution list is created automatically whenever a portal is created. Users added or removed from the portal are implicitly added or removed from the corresponding portal distribution list, provided that the LDAP Base DN does not change (only one LDAP Base DN is supported) and that users created on Microsoft Exchange Active Directory correspond with users created in the identity store used by WebCenter Portal. To disable this feature, do not enter the LDAP (Active Directory) server details in the mail connection.

For more information, see step 7 of Section 15.4.1, "Registering Mail Servers Using Fusion Middleware Control."

For information about adding users on a mail server, see the mail server's product documentation. For information about adding users to WebCenter Portal's identity

store, see Section 31.3, "Adding Users to the Embedded LDAP Identity Store."

Microsoft Exchange 2007 is the only mail server for which there are configuration prerequisites. If you are working with a different mail server (including Microsoft Exchange 2003), then you can bypass the rest of this section.

15.3.2.1 Configuring Microsoft Exchange Server 2007 for WebCenter Portal

The Microsoft Exchange Server 2007 certificate must be added to the WebCenter Portal keystore. This requires the following steps.

1. Section 15.3.2.1.1, "Obtain the Certificate from the Microsoft Exchange Server"
2. Section 15.3.2.1.2, "Add the Certificate to the WebCenter Portal Keystore"
3. Restart the server after the certificate is imported.

15.3.2.1.1 Obtain the Certificate from the Microsoft Exchange Server

Obtain the certificate from your mail server installation administrator. This section describes one way to get the certificate from the Microsoft Exchange Server.

Follow these steps to obtain the certificate from a Microsoft Exchange 2007 server.

1. Open a browser and connect to your IMAP server with the following command:

   ```
   https://host_name/owa
   ```

   Where *host_name* is the name of the Microsoft Exchange Server.

2. Place your cursor on the page, right-click, and select **Properties**, then click **Certificate**.
3. In the popup window, click the **Details** tab, and click **Copy to File...**

   Be sure to use the DER encoded binary (X.509) format, and copy to a file.

4. Convert the .DER format certificate to .PEM format.

   Note: WebLogic only recognizes .PEM format.

   Use Firefox 3.0 or later to download the certificate directly to .PEM format. For other browsers, use the WebLogic Server `der2pem` tool to convert to .PEM format. For more information about `der2pem`, see *Oracle Fusion Middleware Command Reference for Oracle WebLogic Server*.

15.3.2.1.2 Add the Certificate to the WebCenter Portal Keystore

1. Import the downloaded certificate into the keystore, which is generally the file named `cacerts` in the `JAVA_HOME`. For example:

   ```
   keytool -import -alias imap_cer -file cert_file.cer -keystore cacerts
   -storepass changeit
   ```

   Where `cert_file` is the name of the certificate file you downloaded. In a standard installation, the `JAVA_HOME` is in the following location:

   ```
   /scratch/wcinstall/ps2/1225/wlshome/jrockit_160_17_R28.0.0-616
   ```

   See Section 33.4.2.1.3, "Configuring and Exporting the Certificates," for information about adding the certificate to the keystore.

2. Restart the server.

15.3.2.1.3 Microsoft Exchange Server Considerations

- The IMAP port is 993 and secured true. SMTP port is 587 and secured true.

  (Microsoft Exchange Server 2005 used 465.)

- If you see the following error, then you must change the trust store entry in the domain startup file `setDomainEnv.sh`:

```
Caused by: java.io.IOException: Keystore was tampered with, or password was
incorrect
 at sun.security.provider.JavaKeyStore.engineLoad(JavaKeyStore.java:771)
 at sun.security.provider.JavaKeyStore$JKS.engineLoad(JavaKeyStore.java:38)
 at java.security.KeyStore.load(KeyStore.java:1185)
 at com.sun.net.ssl.internal.ssl.TrustManagerFactoryImpl.getCacertsKeyStore
(TrustManagerFactoryImpl.java:202)
 at com.sun.net.ssl.internal.ssl.DefaultSSLContextImpl.getDefaultTrustManager
(DefaultSSLContextImpl.java:70)
```

  To change the entry:

  a. Shutdown the managed server on which WebCenter Portal is deployed.

  b. Edit the domain startup script `setDomainEnv` located at:

     UNIX: *DOMAIN_HOME*`/bin/setDomainEnv.sh`

     Windows: *DOMAIN_HOME*`\bin\setDomainEnv.cmd`

  c. Add the Java property, as follows:

```
-Djavax.net.ssl.trustStore=<path to truststore>
-Djavax.net.ssl.trustStorePassword=<truststore password>
```

     For example:

```
set JAVA_PROPERTIES=
-Dplatform.home=%WL_HOME% -Dwls.home=%WLS_HOME% -Dweblogic.home=%WLS_HOME%
-Djavax.net.ssl.trustStore=C:\jive\mailtool\jssecacerts
-Djavax.net.ssl.trustStorePassword=changeit
```

  d. Restart the managed server.

15.3.3 Mail Server - Security Considerations

For more information, see Section 35.8, "Securing the WebCenter Portal Connection to IMAP and SMTP with SSL."

> **Note:** If LDAP is configured to run in secure mode, then add the `LDAP Secured` property (set to `true/false`) to use LDAP while creating distribution lists. For more information, see Table 15–5.

15.3.4 Mail Server - Limitations

In WebCenter Portal, mail requires a Microsoft Exchange mail server connection to enable automatic WebCenter Portal distribution list management.

15.4 Registering Mail Servers

You can register multiple mail server connections. To start using the new mail connections you must restart the managed server on which WebCenter Portal is deployed.

This section includes the following subsections:

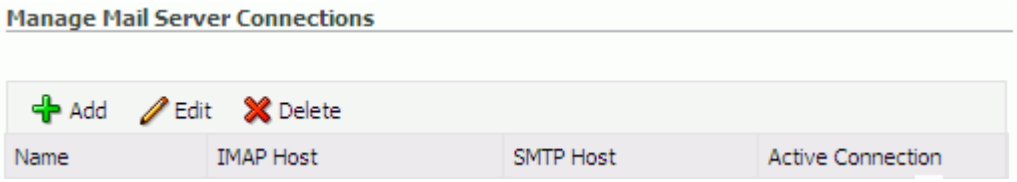
- Section 15.4.1, "Registering Mail Servers Using Fusion Middleware Control"
- Section 15.4.2, "Registering Mail Servers Using WLST"

15.4.1 Registering Mail Servers Using Fusion Middleware Control

To register a mail server with WebCenter Portal:

1. Log in to Fusion Middleware Control and navigate to the home page for the WebCenter Portal. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal > Service Configuration**.
3. On the WebCenter Portal Service Configuration page, select **Mail Server**.
4. To connect to a new mail server, click **Add** (Figure 15–3).

Figure 15–3 Configuring Mail Servers



5. Enter a unique name for this connection, and indicate whether this connection is the active (or default) connection for the application (Table 15–3).

Table 15–3 Mail Server Connection - Name

Field	Description
Connection Name	Enter a unique name for the connection. The name must be unique (across all connection types) within WebCenter Portal.

Table 15–3 (Cont.) Mail Server Connection - Name

Field	Description
Active Connection	Select to indicate whether this connection is the default (or active) connection for mail. You can register multiple mail server connections: ■ **WebCenter Portal** supports multiple mail connections. The mail connection marked *active* is the default connection for mail. All additional connections are offered as alternatives; users can choose which one they want to use through user preferences. ■ **Portal Framework application** only use one mail connection—the connection marked *active*. Any additional connections are ignored.

6. Enter connection details for the mail server (Table 15–4).

Table 15–4 Mail Server Connection Details

Field	Description
IMAP Host	Enter the host name of the computer where IMAP (Internet Message Access Protocol) is running.
IMAP Port	Enter the port on which IMAP listens.
IMAP Secured	Indicate whether a secured connection (SSL) is required for incoming mail over IMAP.
SMTP Host	Enter the host name of the computer where SMTP (Simple Mail Transfer Protocol) is running.
SMTP Port	Enter the port on which SMTP listens.
SMTP Secured	Indicate whether a secured connection (SSL) is required for outgoing mail over SMTP.

Table 15–4 (Cont.) Mail Server Connection Details

Field	Description
Associated External Application	Associate the mail server with an external application. External application credential information is used to authenticate users against the IMAP and SMTP servers. Mail uses the same credentials to authenticate the user on both IMAP and SMTP. You can select an existing external application from the list, or click **Create New** to configure a new external application. For more information, see Chapter 23, "Managing External Applications." The external application for mail must use `Authentication Method=POST`, and you can customize some mail header fields (with **Display to User** enabled): ▪ Property: `mail.user.emailAddress` (who the mail is from) Property: `mail.user.displayName` (display name from the mail) Property: `mail.user.replyToAddress` (address used to reply to the mail) These properties ensure that a specific mail address is the same in the external application and in the mail server. They are added to the mail connection and are used by mail for the **From**, **Display Name** and **Reply To** fields (Figure 15–4). See Table 15–8 for Additional Properties configuration. If your application offers a self-registration page with the facility to mail user ID information on request, then you must ensure that public credentials are configured for the external application selected here. If public credentials are not defined, then mail cannot be sent to users on their request. WebCenter Portal, for example, offers this feature on its default self-registration page.

7. Specify LDAP connection details for the Active Directory server managing WebCenter Portal distribution lists (Table 15–5).

   WebCenter Portal supports Microsoft Exchange where distribution lists are managed on an Active Directory server.

   Note: Active Directory server details must be provided as part of the mail connection for *distribution lists* to work in WebCenter Portal.

Table 15–5 LDAP Directory Server Configuration Parameters

Field	Description
LDAP Host	Enter the host name of the computer where the LDAP directory server (Lightweight Directory Access Protocol) is running.
LDAP Port	Enter the port on which the LDAP directory server listens.
LDAP Base DN	Enter the base distinguished name for the LDAP schema. For example, `CN=Users,DC=oracle,DC=com`.
LDAP Domain	Enter the domain appended to distribution list names. For example, if the domain value is set to `example.com`, then a portal named Finance Project maintains a distribution list named `FinanceProject@example.com`.

Table 15–5 (Cont.) LDAP Directory Server Configuration Parameters

Field	Description
LDAP Administrator User Name	Enter the user name of the LDAP directory server administrator. A valid user with privileges to make entries into the LDAP schema.
LDAP Administrator Password	Enter the password for the LDAP directory server administrator. The password is stored in a secured store.
LDAP Default User	Enter a comma-delimited list of user names to whom you want to grant moderation capabilities. These users become members of every portal distribution list that is created. The users specified must exist in the base LDAP schema (specified in the `LDAP Base DN` field).
LDAP Secured	Indicate whether a secured connection (SSL) is required between WebCenter Portal and the LDAP directory server.

8. Configure advanced options for the mail server connection (Table 15–6).

Table 15–6 Mail Server Connection - Advanced Configuration

Field	Description
Connection Timeout (seconds)	Specify a suitable timeout for the connection. This is the length of time (in seconds) WebCenter Portal waits for a response from the mail server before issuing a connection timeout message. The default is -1, which means that the default is used. The default is 10 seconds.

9. Optionally, you can add more parameters to the mail server connection (Table 15–7).

Table 15–7 Additional Mail Connection Properties

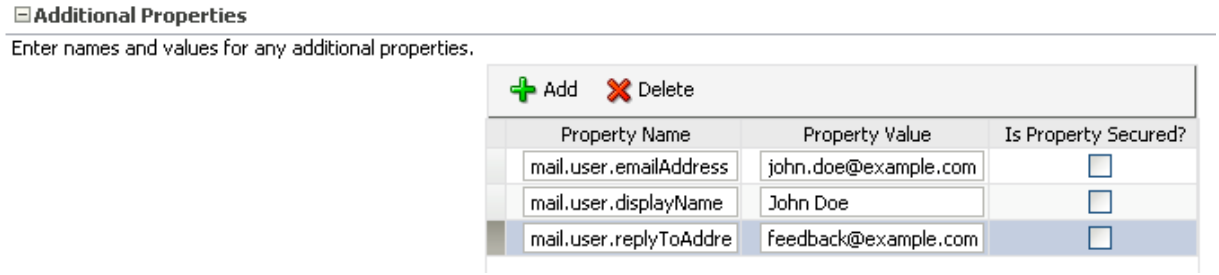
Additional Connection Property	Description
`charset`	Characterset used on the connection. The default charset is UTF-8. To use a different characterset, such as ISO-8859-1, set the charset connection property.
Various IMAP properties	Any valid IMAP connection property. For example, `mail.imap.connectionpoolsize`. A list of valid IMAP properties are listed in documentation for the `com.sun.mail.imap` package at: `http://javamail.java.net/nonav/docs/api`
Various SMTP properties	Any valid SMTP connection property. For example, `mail.smtp.timeout`. A list of valid SMTP properties are listed in the documentation for the `com.sun.mail.smtp` package at: `http://javamail.java.net/nonav/docs/api`

If additional parameters are required to connect to the mail server, expand **Additional Properties** and enter details as required (see Table 15–8).

Table 15–8 Mail Connection - Additional Properties

Field	Description
Add	Click **Add** to specify an additional connection parameter: ▪ **Property Name** -Enter the name of the connection property. ▪ **Property Value** - Enter the default value for the property. ▪ **Is Property Secured** - Indicate whether encryption is required. When selected, the property value is stored securely using encryption. For example, select this option to secure the `admin.password` property where the value is the actual password.
Delete	Click **Delete** to remove a selected property. Select the correct row before clicking **Delete**. **Note:** Deleted rows appear disabled until you click **OK**.

Figure 15–4 Additional Properties for Mail Connection



10. Click **OK** to save this connection.
11. To start using the new (active) connection you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed.

    For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

15.4.2 Registering Mail Servers Using WLST

Use the WLST command `createMailConnection` to create a mail server connection. For command syntax and examples, see the "createMailConnection" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

Use the WLST command `setMailConnectionProperty` to add additional required properties through your external application. The external application for mail must use Authentication Method=POST, and you can customize some mail header fields (with Display to User enabled). For example:

```
setMailConnectionProperty(appName='webcenter', name='NotificationSharedConn',
key='mail.user.emailAddress', value='john.doe@example.com')

setMailConnectionProperty(appName='webcenter', name='NotificationSharedConn',
key='mail.user.displayName', value='John Doe')

setMailConnectionProperty(appName='webcenter', name='NotificationSharedConn',
key='mail.user.replyToAddress', value='feedback@example.com')
```

where:

- `mail.user.emailAddress` = Email Address ('From' from the mail)
- `mail.user.displayName` = Your Name (display name from the mail)
- `mail.user.replyToAddress` = Reply-To Address (address when replying to the mail)

These properties ensure that a specific mail address is the same in the external application and in the mail server. These properties are added to the Mail connection and are used by mail for the From, Display Name and Reply To fields.

For Exchange 2007 only, create an universal distribution list which means that the default property value of `2` should be updated to `8`. Specify a value of `8` for the mail property `mail.exchange.dl.group.type`, as follows:

```
setMailServiceProperty(appName='webcenter',
property='mail.exchange.dl.group.type', value='8')
```

If your application offers a self-registration page with the facility to mail user ID information on request, then you must ensure that public credentials are configured for the external application selected here. If public credentials are not defined, then mail cannot be sent to users on their request. WebCenter Portal, for example, offers this feature on its default self-registration page.

For command syntax and examples, see the "setMailConnectionProperty" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

To configure mail to use the new mail server connection as its default connection, set `default=1` (true). For more information, see Section 15.5.2, "Choosing the Active (or Default) Mail Server Connection Using WLST."

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

> **Note:** To start using new connections you must restart the managed server on which WebCenter Portal or your Portal Framework application is deployed. For more information, see the "Starting and Stopping Managed Servers Using the Command Line" section in *Oracle Fusion Middleware Administrator's Guide*.

15.5 Choosing the Active (or Default) Mail Server Connection

You can register multiple mail server connections with WebCenter Portal, but only one connection can be designated as the default connection. The *default connection* becomes the back-end mail server for:

- Mail task flows
- WebCenter Portal distribution lists
- Anywhere there is a **Send Mail** icon

This section includes the following subsections:

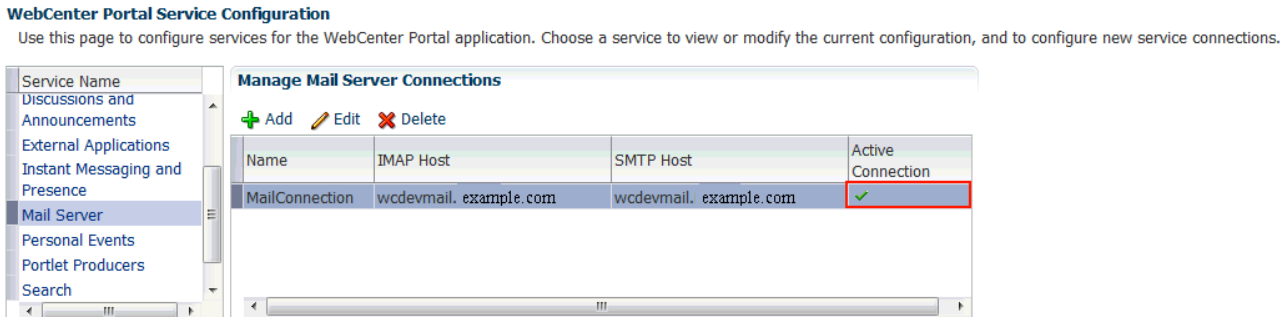
15.5.1 Choosing the Active (or Default) Mail Server Connection Using Fusion Middleware Control

To change the default connection:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal > Service Configuration**.
3. On the WebCenter Portal Services Configuration page, select **Mail Server**.

   The Manage Mail Server Connections table indicates the current active connection, if any (Figure 15–5).

Figure 15–5 Mail Server - Active Connection



4. Select the connection you want to make the active (or default) connection, and then click **Edit**.
5. Select the **Active Connection** check box.
6. Click **OK** to update the connection.
7. To start using the new default connection you must restart the managed server on which WebCenter Portal is deployed. For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

15.5.2 Choosing the Active (or Default) Mail Server Connection Using WLST

Use the WLST command `setMailConnection` with `default=1` (true) to make an existing mail server connection the default connection for mail. For command syntax and examples, see the "setMailConnection" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

A connection does not cease to be the default connection for mail if you change the default argument from `0` to `1` (`true` to `false`).

To disable a mail connection, either delete it, make another connection the 'active connection', or use the `removeMailServiceProperty` command:

```
removeMailServiceProperty(appName='webcenter', property='selected.connection')
```

Using this command, connection details are retained but the connection is no longer named as an active connection. For more information, see the "removeMailServiceProperty" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Note: To start using the active connection you must restart the managed server on which WebCenter Portal is deployed. For more information, see the "Starting and Stopping Managed Servers Using the Command Line" section in *Oracle Fusion Middleware Administrator's Guide*.

15.6 Modifying Mail Server Connection Details

You can modify mail server connection details at any time.

To start using updated mail connections you must restart the managed server on which WebCenter Portal is deployed.

This section includes the following subsections:

- Section 15.6.1, "Modifying Mail Server Connection Details Using Fusion Middleware Control"
- Section 15.6.2, "Modifying Mail Server Connection Details Using WLST"

15.6.1 Modifying Mail Server Connection Details Using Fusion Middleware Control

To update mail server connection details:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Framework applications - From the **Application Deployment** menu, select **WebCenter Portal > Service Configuration**.
3. On the WebCenter Portal Service Configuration page, select **Mail Server**
4. Select the connection name, and click **Edit**.
5. Edit connection details, as required.

   For detailed parameter information, see Table 15–4, " Mail Server Connection Details".
6. Click **OK** to save your changes.

7. To start using updated connection details you must restart the managed server on which WebCenter Portal is deployed. For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

15.6.2 Modifying Mail Server Connection Details Using WLST

Use the WLST command `setMailConnection` to edit existing mail server connection details. For command syntax and examples, see the "setMailConnection" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

If additional parameters are required to connect to your mail server, use the `setMailConnectionProperty` command. For more information, see the "setMailConnectionProperty" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Note: To start using the updated connections you must restart the managed server on which WebCenter Portal is deployed. For more information, see the "Starting and Stopping Managed Servers Using the Command Line" section in *Oracle Fusion Middleware Administrator's Guide*.

15.7 Deleting Mail Server Connections

You can delete mail server connections at any time, but use caution when deleting the active (or default) connection. If you delete the active connection, Mail task flows do not work, as they all require a back-end mail server.

When you delete a connection, consider deleting the external application associated with the mail server connection *if* the application's sole purpose was to support this connection. For more information, see Section 23.6, "Deleting External Application Connections."

This section includes the following subsections:

- Section 15.7.1, "Deleting a Mail Connection Using Fusion Middleware Control"
- Section 15.7.2, "Deleting a Mail Connection Using WLST"

15.7.1 Deleting a Mail Connection Using Fusion Middleware Control

To delete a mail server connection:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.

- For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal > Service Configuration**.

3. On the WebCenter Portal Services Configuration page, select **Mail Server**.
4. Select the connection name, and click **Delete**.
5. To make this change you must restart the managed server on which WebCenter Portal is deployed. For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

Note: Before restarting the managed server, mark another connection as active; otherwise, mail is disabled.

15.7.2 Deleting a Mail Connection Using WLST

Use the WLST command `deleteConnection` to remove a mail server connection. For command syntax and examples, see the "deleteConnection" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

15.8 Setting Up Mail Defaults

Use the WLST command `setMailServiceProperty` to set defaults for mail:

- `address.delimiter`: Defines the delimiter that is used to separate multiple mail addresses. A comma is used by default.

  Some mail servers require mail addresses in the form `lastname, firstname` and, in such cases, a semicolon is required.

- `mail.emailgateway.polling.frequency`: Frequency, in seconds, that portal distribution lists are checked for new incoming mail messages. The default is 1800 seconds (30 minutes).

  Email communication through WebCenter Portal distribution lists can be published as discussion forum posts on a discussions server. For details, see the "Publishing Portal Mail in a Discussion Forum" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

- `mail.messages.fetch.size`: Maximum number of messages displayed in mail inboxes

- `resolve.email.address.to.name`: Determines whether user email addresses are resolved to WebCenter Portal user names when LDAP is configured. Valid values are `1` (`true`) and `0` (`false`). The default value is `0`.

  When set to `1`, WebCenter Portal user names display instead of email addresses in Mail task flows.

  Set this property to `1` if instant messaging and presence requires user names to obtain presence status because presence information cannot be obtained when mail provides email addresses. Setting this value to `1` does impact application performance so you must take this into consideration when setting this property.

- `mail.recipient.limit`: Restricts the number of recipients to a message. For example, setting this value to '500' limits the number of recipients to 500.

For command syntax and examples, see the "setMailServiceProperty" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

15.9 Testing Mail Server Connections

Confirm that the mail server is running by connecting to the server using any client, such as Thunderbird or Outlook.

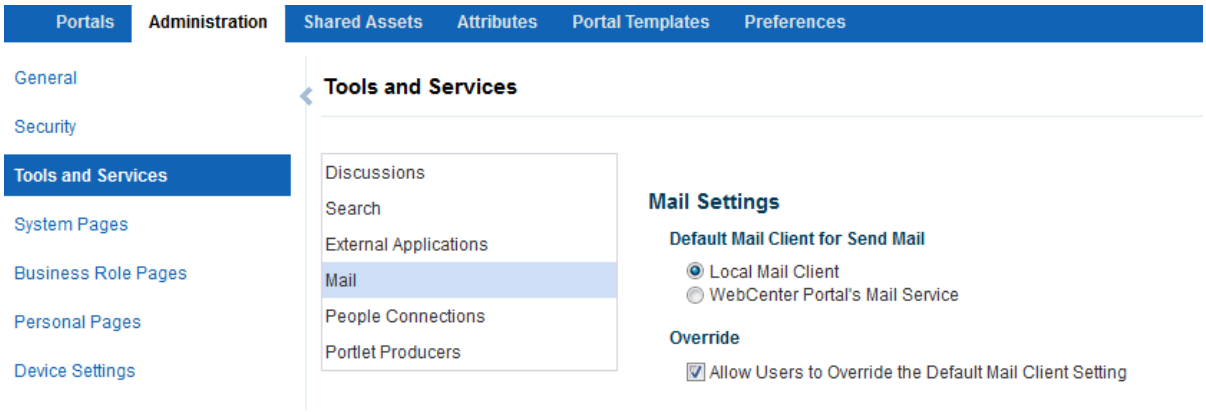
For Microsoft Exchange, go to **Administrative Tools - Services** to confirm that the following components are running (Status: Started):

- Microsoft Exchange IMAP4
- Simple Mail Transfer Protocol (SMTP)

15.10 Configuring Send Mail Notifications for WebCenter Portal

System administrators are responsible for setting mail options through WebCenter Portal administration settings (Figure 15–6).

Figure 15–6 Setting Mail Options



From this page, you can assign the mail client for the Send Mail feature. This feature allows application assets to send mail directly from their task flows, using the **Send Mail** icon (Figure 15–7).

Figure 15–7 Send Mail Icon

For example, from an announcement, users can click the **Send Mail** icon to open a mail window prepopulated with information including the announcement text, author, date created, and location. They can edit and add to the mail, as necessary. The way the mail window is prepopulated depends on the resource sending it. For example, from a wiki, Send Mail opens a mail window prepopulated with the name of the wiki, the size, who created it and when, who modified it and when, and a URL link to the wiki.

Within a portal, the mail can be addressed to all members of the portal, which is the default distribution list that is created when the portal is created. Moderators (and anyone granted the `Manage Configuration` permission on the portal) set this through the Tools and Services page in the portal's administration settings. See the "Configuring the Mail Distribution List for a Portal" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

For all Send Mail notifications throughout WebCenter Portal, you can choose to use the local mail client, such as Microsoft Outlook or Mozilla Thunderbird, or WebCenter Portal's own Mail service. The local mail client is the default. The Send Mail feature does not require the Mail service, that is, if the Mail service is not yet configured, you can still use the Send Mail feature with WebCenter Portal's Mail service. Application specialists or portal moderators can specify whether portal participants can override the default mail client setting.

Note: With some browsers, Send Mail notifications are garbled for many non-English languages. When multibyte characters are encoded (required for the "mailto:" protocol), the URL length exceeds the browser limit. As a workaround, configure the Send Mail feature to use WebCenter Portal's Mail service instead of the local mail client.

As the system administrator, you can also specify whether users can override the default mail client setting.

See Also: Section 15.4, "Registering Mail Servers"

15.10.1 Enabling Shared Mail Connections for Send Mail Notifications

Users do not need to specify credentials when sending mail using WebCenter Portal's Mail service when *shared credentials* are configured for the external application that is associated with the mail server connection.

To enable shared mail connections:

1. Confirm that your portal is using WebCenter Portal's Mail service to send mail.
   a. Open the Administration tab.
   b. Click **Tools and Services**, and then select **Mail**.
   c. Ensure that **Default Mail Client for 'Send Mail'** is set to **WebCenter Portal's Mail Service**.

   See also Section 15.10, "Configuring Send Mail Notifications for WebCenter Portal."
2. Set up a mail connection that uses an external application configured with the shared credentials and record the mail connection name.
3. Open the portal where shared mail credentials are required and specify the name of the shared mail connection.

   For details, see the "Configuring a Shared Mail Connection for a Portal" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

15.11 Troubleshooting Issues with Mail

This section includes the following subsections:

15.11.1 Mail is Not Accessible in Secure Mode

Problem

You configured mail to function in secure mode, but it is not accessible.

Solution

Ensure the following:

- IMAP and SMTP ports are specified correctly. See Section 15.4, "Registering Mail Servers."
- Properties are set to `true` in your mail server.
  - `mail.imap.secured = true`
  - `mail.smtp.secured = true`

15.11.2 Mail is Not Accessible in Non-Secure Mode

Problem

You configured mail to function in non-secure mode, but it is not accessible.

Solution

Ensure the following:

- IMAP and SMTP ports are specified correctly. See, Section 15.4, "Registering Mail Servers."
- Properties are set to `false` in your mail server.
  - `mail.imap.secured = false`
  - `mail.smtp.secured = false`

15.11.3 Unable to Create Distribution Lists in the Non-Secure Mode

Problem

You are unable to create portal distribution lists in non-secure mode; that is, SSL is not configured on the LDAP server.

Solution

Check if the mail server has been reinstalled or the user has been deleted. Also ensure that the following parameters are configured accurately in non-secure mode, in the LDAP server:

- `ldapHost`
- `defaultUser`
- `ldapAdminPassword`
- `ldapBaseDN`
- `ldapPort`

See Section 15.4, "Registering Mail Servers."

15.11.4 Unable to Create Distribution Lists in the Secure Mode

Problem

You are unable to create WebCenter Portal distribution lists in secure mode, that is, SSL is configured on the LDAP server.

Solution

Check if the mail server has been reinstalled or the user has been deleted. Also ensure that the following parameters are configured accurately in secure mode, in the LDAP server:

- `ldapHost`
- `defaultUser`
- `ldapAdminPassword`
- `ldapBaseDN`
- `ldapPort`
- `ldap.connection.secure, 'true'`

See Also: Section 15.4, "Registering Mail Servers"

15.11.5 Provisioning of Mail Fails in a Portal (Default Distribution List not Created)

Problem

In WebCenter Portal, when accessing a portal's Tools and Services Mail page, the following error message appears "*Provisioning of Mail service for this portal has failed*" and the Distribution List field is blank.

Solution

Make sure that the portal name is unique. If the portal name is not unique, a distribution list already exists. You can select the existing distribution list or select another distribution list.

15.11.6 Unable to Configure the Number of Mail Messages Downloaded

Problem

You cannot configure how many mail messages are downloaded to each user's Inbox.

Solution

Use the `setMailServiceProperty` WLST command. For example, to download 100 mail messages from the mail client, specify the `mail.messages.fetch.size` parameter as `100`, as shown in the following example:

```
setMailServiceProperty(appName='webcenter',  property='mail.messages.fetch.size',
value='100')
```

For command syntax and examples, see the "setMailServiceProperty" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

15.11.7 Unable to Publish and Archive WebCenter Portal Mail

Problem

You are unable to archive WebCenter Portal mail.

Solution

If the archiving fails, check the following:

- In WebCenter Portal, navigate to **Administration** > **Tools and Services** > **Discussions**. Check whether the required configuration is accurate. For details, see the "Publishing Portal Mail in a Discussion Forum" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.
- Check whether the user account configured here is a member of the distribution list.
- For a particular portal, check whether the forum configured is available in the discussions server. For details, see the "Publishing Portal Mail in a Discussion Forum" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.
- Check whether the user who sends mail to the distribution list is available in the discussions server and the mail address is the same.

15.11.8 Changing Passwords on Microsoft Exchange

Problem

If multiple users log on to Microsoft Exchange with the same user name and password, and then one user changes the password, the original password remains valid until all users log off.

For example, say the current password of the user `monty` is `mypassword`. Two users, A and B, log on from different clients using WebCenter Portal or Microsoft Exchange. Both log on as `monty/mypassword`, and both are able to see the mail messages. Now user A changes the password in Microsoft Exchange to `oracle1`. Because there currently are clients using the passwords `oracle1` and `mypassword`, both are valid passwords; that is, new users can log on as `monty/mypassword` and still see the mail messages.

Solution

After all existing users with the original password log off, the new password takes effect. Until then, users can use both passwords to log on.

15.11.9 Mail Content Sent as Attachments

Problem

When users receive mail in Framework applications, message content is shown as an attachment (named `content.html`) rather than within the message body. This can occur if the mail server is running Microsoft Exchange Server 2003 and the "*Update Rollup 3 for Microsoft Exchange Server 2007*" is not yet installed.

Solution

Download and install "*Update Rollup 3 for Microsoft Exchange Server 2007*" which fixes this issue. For more information, see `http://support.microsoft.com/kb/930468`.

16 Managing People Connections

This chapter describes back-end configuration requirements for the People Connections service.

This chapter includes the following sections:

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console and the `Administrator` role in the deployed application:

- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.
- **Portal Framework application**: `Administrator` role granted through the Administration Console.

For more information about roles and permissions, see Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

16.1 About the People Connections Service

The People Connections service provides social networking tools for creating, interacting with, and tracking the activities of one's connections. Its features enable users to manage their personal profiles, access the profiles of other users, provide *ad hoc* feedback, post messages, track activities, and connect with others.

It does this through a set of features that include:

- **Activity Stream** for viewing user activities generated through application or social networking actions.
- **Connections** for connecting to other application users to share information, comment on performance, exchange messages, and track activity
- **Feedback** for giving ad hoc performance feedback to other users
- **Message Board** for posting messages to other users
- **Profile** for entering personal contact information and viewing the contact information of other users
- **Publisher** for publishing status messages and posting files and links

The features of the People Connections service fall into the above five categories. Each category includes a set of task flows that expose People Connections features to end users.

> **See Also:** For information on adding People Connections functionality to a portal, see the "Adding Connections to a Portal" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Always use the Fusion Middleware Control or WLST command-line tool to review and configure back-end services for WebCenter Portal and Portal Framework applications. Any changes you make to WebCenter Portal or Portal Framework applications, post deployment, are stored in MDS metadata store as customizations. See Section 1.3.5, "Oracle WebCenter Portal Configuration Considerations."

> **Note:** Most changes you make to WebCenter Portal tools and services configuration through Fusion Middleware Control or using WLST are not dynamic. For your changes to take effect, you must restart the managed server in which the application is deployed. See Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

For additional information about configuring People Connections, refer to Section 16.3, "Configuring People Connections for WebCenter Portal."

16.2 People Connections Prerequisites

To use the People Connections service, you must have the WEBCENTER schema installed in your database.

In a production environment, an enterprise can leverage its back-end identity store as a means of providing People Connections with a population of potential connections. In a development environment, developers can add test-users to the `jazn-data.xml` file.

For example, Profile takes the bulk of its information from the back-end identity store that provides WebCenter Portal or your Portal Framework application with its users. Additionally, Profile may offer opportunities for altering some of this information and for providing additional data not included in the identity store.

For information about connecting to a back-end (LDAP) identity store for the production version of your application, see Chapter 31, "Configuring the Identity Store."

16.3 Configuring People Connections for WebCenter Portal

This section steps you through the process of setting application-wide values for People Connections features. It includes the following subsections:

- Section 16.3.1, "Accessing People Connections Administrative Settings"
- Section 16.3.2, "Configuring Activity Stream"
- Section 16.3.3, "Configuring Connections"
- Section 16.3.4, "Configuring Profile"
- Section 16.3.5, "Configuring Message Board"
- Section 16.3.6, "Configuring Feedback"

16.3.1 Accessing People Connections Administrative Settings

To access People Connections administrative settings:

1. Open WebCenter Portal Administration.

   For details, see Section 47.2, "Accessing the Portal Builder Administration Page."

2. Click **Tools and Services**, and then select **People Connections**.

   Alternatively, use the following URL, and then select **People Connections**:

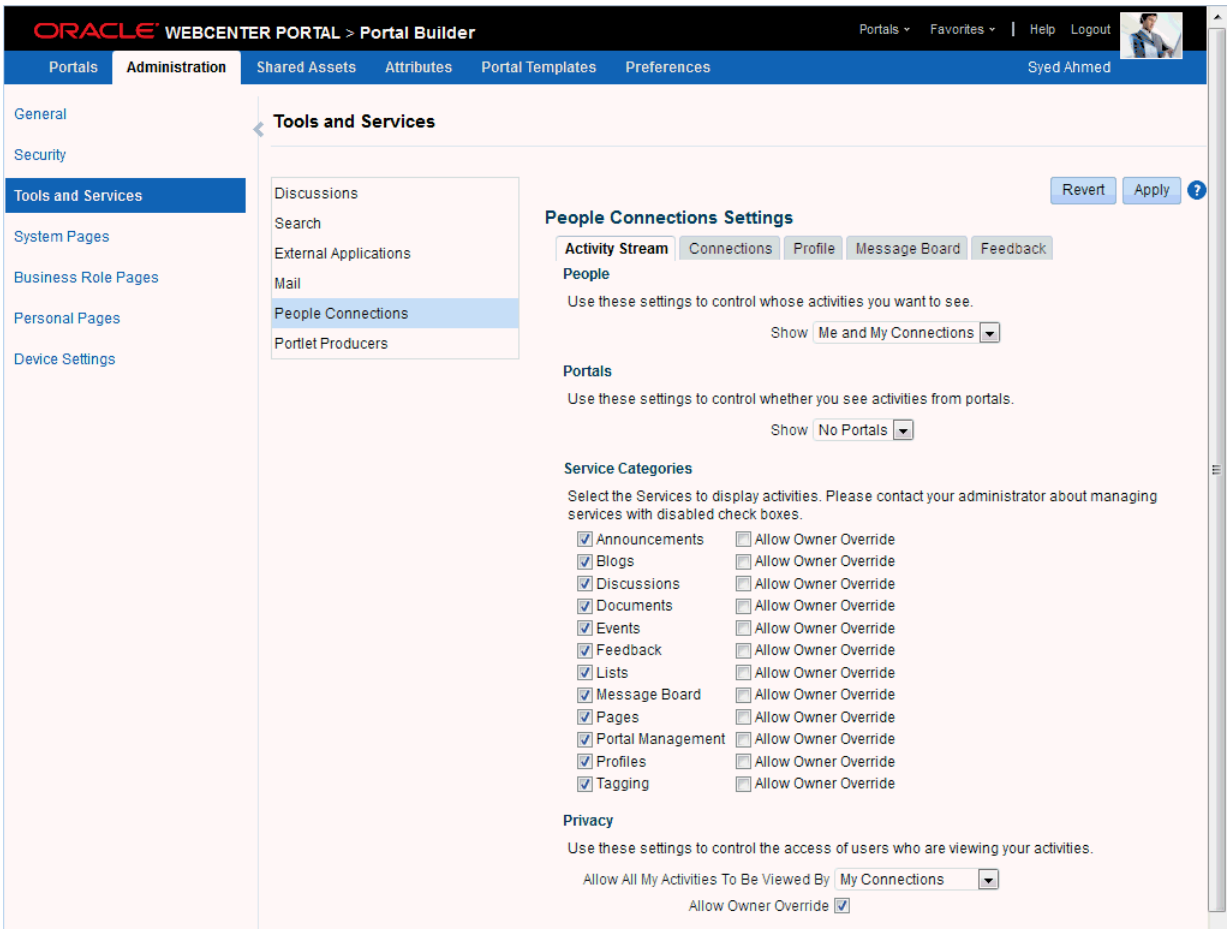
   `http://host:port/webcenter/portal/admin/tools`

   Tabs with the names of People Connections features appear to the right.

16.3.2 Configuring Activity Stream

Activity Stream provides a means of publishing and tracking users' application activity. Activity Stream configuration settings specify from which users and activities are streamed, who can see a user's streamed activities, and whether liking and commenting is available on each streamed activity (Figure 16–1).

Figure 16–1 Administration Settings for People Connections



Who can view a user's activities and the types of activities tracked depend on Activity Stream configuration. Table 16–1 lists the types of activities that may be tracked by Activity Stream.

Table 16–1 Activities Tracked by Activity Stream

Feature Area	Tracked Activities	Scope	Activities Shared or Private
Announcements	▪ Create announcement ▪ Edit announcement	Portal	Shared with other portal members
Blogs	▪ Create blog ▪ Update blog	▪ Portal ▪ Home portal	▪ Activities on portal blogs are shared with other portal members. ▪ Activities on Home portal blogs are shared with the blogger's connections.
Connections	▪ Invitations to connect ▪ People are connected	Home portal	Shared with invitor and invitee's connections
Discussions	▪ Create forum ▪ Create topic ▪ Reply to topic	Portal	Shared with other portal members

Table 16–1 (Cont.) Activities Tracked by Activity Stream

Feature Area	Tracked Activities	Scope	Activities Shared or Private
Documents	■ Create document ■ Edit document ■ Add tag ■ Remove tag	■ Portal ■ Home portal	■ Activities on portal documents are shared with other portal members. ■ Activities on Home portal documents are private to user.
Events	■ Create an event ■ Edit an Event	Portal	Shared with other portal members
Feedback	■ Feedback left ■ Feedback received	Home portal	Shared with whomever is permitted to view such activities. (For more information, see the "Setting Feedback Preferences" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.)
Lists	■ Create a list ■ Add a row to a list ■ Edit a list row	Portal	Shared with other portal members
Message Board	■ Message left ■ Message received	Home portal	Shared with whomever is permitted to view such activities. (For more information, see the "Setting Message Preferences" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.)
Pages	■ Create page ■ Edit page ■ Add tag ■ Remove tag	■ Portal ■ Home portal	■ Activities on portal pages are shared with other portal members. ■ Activities on Home portal pages are private to user.
Profiles	■ Photo updated ■ Profile updated ■ Personal status note updated	Home portal	Shared with whomever is permitted to view such activities. (For more information, see the "Setting Profile Preferences" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal.)*
WebCenter Portal Management	■ Create portal ■ Join portal	Portal	Shared with other portal members
Tagging	■ Add tag ■ Remove tag	■ Portal ■ Home portal	■ Activities in a portal are shared with all portal members. ■ Activities in a Home portal are shared with whomever is permitted to view such activities. (For more information, see Section 16.3.2, "Configuring Activity Stream" and the "Setting Activity Stream Preferences" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.)

Activity Stream configuration falls under the following categories:

- **People**—For determining whose activities to show, either the current user's or both the current user and the user's connections.

- **WebCenter Portal**—For determining whether to show activities from all available portals or just the Home portal.
- **Service Categories**—For selecting the services from which to report activities and enabling users to override these default selections in their personal preferences or preventing users from overriding.
- **Privacy**—For selecting who may see the current user's activities.
- **Comments and Likes**—For enabling users to comment on a posted activity and express a liking for a posted activity

To configure Activity Stream for all users:

1. Open WebCenter Portal Administration.

   For details, see Section 47.2, "Accessing the Portal Builder Administration Page."

2. Click **Tools and Services**, and then select **People Connections**.

   Alternatively, use the following URL, and then select **People Connections**:

   `http://host:port/webcenter/portal/admin/tools`

   Tabs with the names of People Connections features appear to the right.

3. Click the **Activity Stream** tab to bring it forward.
4. Under **People**, select whose activities to show:
   - **Only Me**—Show only the current user's activities in his or her view of the Activity Stream.
   - **Me and My Connections**—Show the current user's activities and the activities of that user's connections in his or her view of the Activity Stream.
   - **No Personal**—Omit all activities streamed from Home portals in the current user's view of his or her Activity Stream.
5. Under **Portals**, select to show activities from:
   - **All Portals**—All portals to which the user has access
   - **My Portals**—All portals of which the user is the moderator
   - **No Portals**—Only the Home portal
6. Under **Service Categories**, select the services from which to publish activity.

   Note: The activities of services that are not selected are still tracked, but they do not appear in the Activity Stream. If you select to show the activities at some later point, then all of the activities that occurred when it was not selected will appear in the Activity Stream.

   Table 16–1 lists the activities tracked by the Activity Stream.

7. Optionally, select **Allow Owner Override** to enable users to override a setting for a given service through their personal preferences.

   Deselect this check box to prevent users from overriding the application defaults you set here.

8. Under **Privacy**, specify who can view the current user's activities and whether users can override this setting in their personal preferences.

   Table 16–2 lists and describes each option.

Table 16–2 Activity Stream Privacy Options

Option	Description
Allow all of my activities to be viewed by	Specify who can view another user's activities. Choose from: ■ **Everyone**—Any user, whether logged in or not, can view other users' activities. ■ **Authenticated Users**—Users who have logged in can view other users' activities. ■ **My Connections**—User A can view user B's activities if user B has accepted user A as a connection. User A can also view user A's activities. ■ **Myself**—Only user A can view user A's activities.
Allow Owner Override	Enable users to override the application default settings using their own People Connections Preferences (for more information, see the "Setting Activity Stream Preferences" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*).

9. Expand the **Likes and Comments** node, and specify whether liking and commenting are allowed:
   - Select **Enable comments on objects in the Activity Stream** to enable users to comment on a given Activity Stream item. Deselect the check box to prevent users from commenting.
   - Select **Enable others to like objects in the Activity Stream** to enable users to express a liking for an Activity Stream item. Deselect the check box to prevent users from commenting.

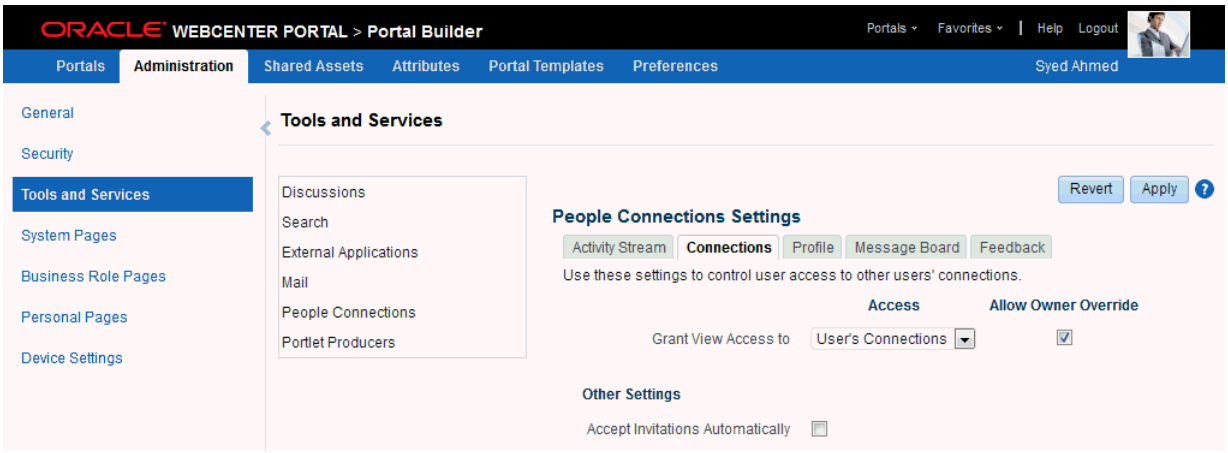
     Tip: Users can like and comment on streamed items that include objects. For example, users can like or comment on "Jack has updated doc.xml," but cannot like or comment on "Jack and Jill are now connected."

     For more information, see the "Liking, Commenting On, and Sharing Items in WebCenter Portal" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.
10. Click **Apply** to save your configuration settings.

16.3.3 Configuring Connections

Connections configuration involves specifying who can view another user's connections and whether users accept invitations to connect automatically (Figure 16–2).

Figure 16–2 Configuration Settings for Connections



To configure Connections:

1. Open WebCenter Portal Administration.

   For details, see Section 47.2, "Accessing the Portal Builder Administration Page"

2. Click **Tools and Services**, and then select **People Connections**.

   Alternatively, use the following URL, and then select **People Connections**:

   `http://host:port/webcenter/portal/admin/tools`

   Tabs with the names of People Connections features appear to the right.

3. Click the **Connections** tab to bring it forward.

   Table 16–3 lists and describes each option.

Table 16–3 Connections Configuration Options

Option	Description
Grant View Access to	Classes of users to whom to grant automatic view access to a user's connections The users you select can view and interact with another user's connections. Choose from: ▪ **Everyone**—All users, including users who are not logged in, can see other users' connections. ▪ **Authenticated users**—Only users who are logged in can see other users' connections. ▪ **User's Connections**—Only the user and the user's connections can see the user's connections. ▪ **User Only**—Only a user can see his or her own connections.
Allow Owner Override	Option to allow or prohibit users from overriding the administrator View access setting: ▪ Select to allow users to override the administrative View access setting specified here using their personal Preferences (for more information, see the "Setting Connections Preferences" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*). ▪ Deselect to prohibit users from overriding the administrative View access setting.

Table 16–3 (Cont.) Connections Configuration Options

Option	Description
Accept Invitations Automatically	▪ Select to specify that, by default, all invitations to connect are accepted automatically. ▪ Deselect to specify that, by default, a user must explicitly accept or reject invitations to connect.

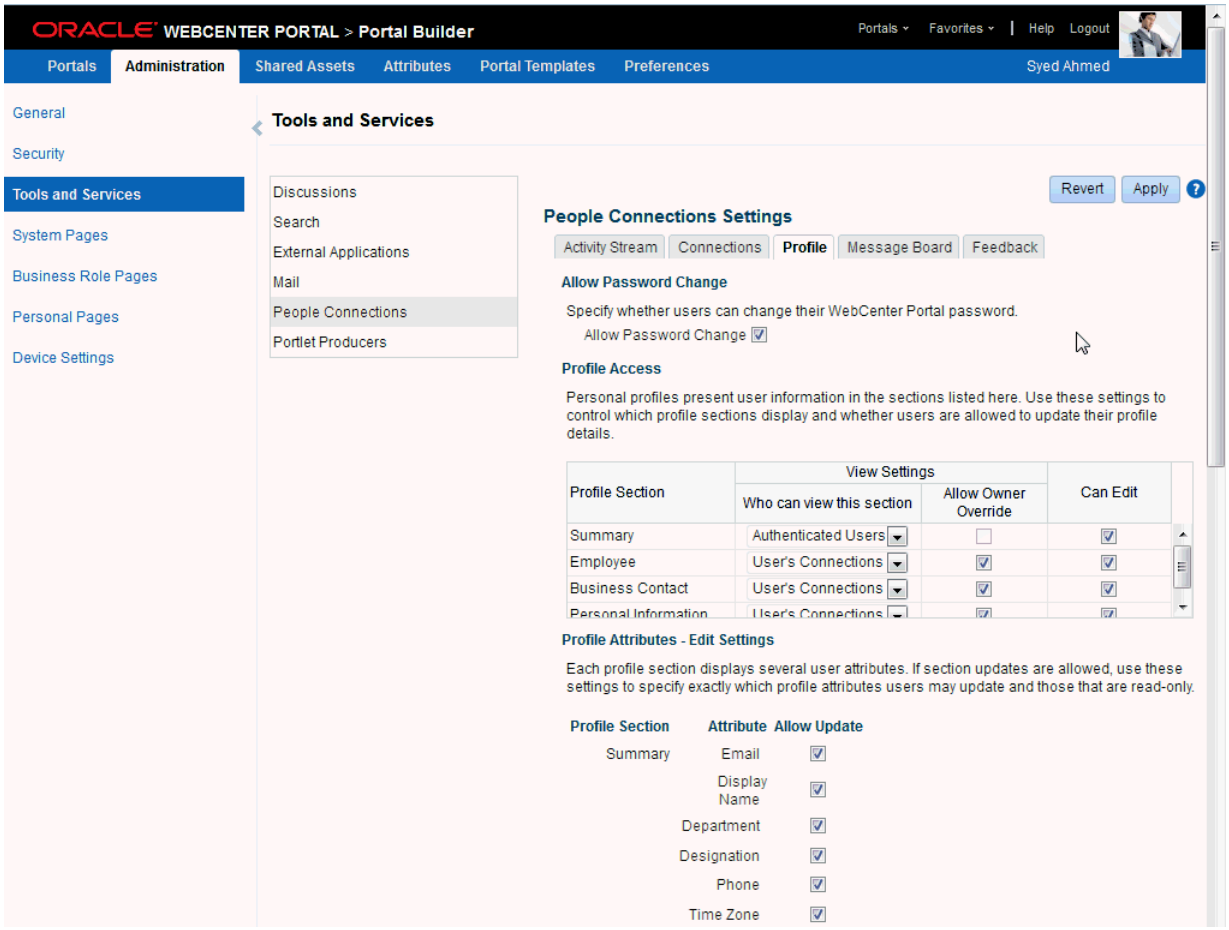
4. Click **Apply** to save your configuration settings.

16.3.4 Configuring Profile

Every authenticated user has a profile that displays personal information, such as the user's email address, phone number, office location, department, manager, direct reports, and so on. All but three attributes are stored and read from the LDAP identity store that is configured for the WebCenter Portal. The three exceptions include the Profile photo and expertise and Publisher status messages.

Use administrative configuration settings for Profile to specify whether users are allowed to change their application passwords, which profile sections display, whether users are allowed to update their profile details, and the profile attributes that users may update (Figure 16–3).

Figure 16–3 Configuration Settings for Profile



Personal profiles are presented in four sections: **Summary**, **Employee**, **Business Contact**, **Personal Information**. Each section provides information related to the section heading. For example, **Summary** includes a collection of basic details, such as the user's name, email address, and office location.

In configuration settings, the access setting for the **Summary** section controls who can search for the user (for example, through global search, people pickers, and the searches one uses to find and invite other users to connect). For example, if Everyone is allowed to view the **Summary** section, then the user can be searched for by unauthenticated (public) users. If only Authenticated Users can view another user's **Summary** section, then only logged in users can search for the user. If None is the selected value for **Who can view this section**, then the user will not appear in search results.

The **Summary** section is the only Profile section for which the privacy setting cannot be changed by the end-user through Preferences (for more information, see the "Setting Profile Preferences" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*). It is controlled at a global level for all users through the settings described in this section.

It is the administrator's job to specify the information to show in each section and determine whether users are allowed to edit their profile data and their application password within the WebCenter Portal.

To configure Profile:

1. Open WebCenter Portal Administration.

   For details, see Section 47.2, "Accessing the Portal Builder Administration Page"

2. Click **Tools and Services**, and then select **People Connections**.

   Alternatively, use the following URL, and then select **People Connections**:

   `http://host:port/webcenter/portal/admin/tools`

   Tabs with the names of People Connections features appear to the right.

3. Click the **Profile** tab to bring it forward.

   Table 16–4 lists and describes each option.

Table 16–4 Profile Configuration Options

Option	Description
Allow Password Change	Specify whether users are allowed to change their application password ■ Select to enable users to change their application password. ■ Deselect to prevent users from changing their application password. This option is useful when your organization provides a single, separate application for managing user credentials and, consequently, prefers not to offer password management through each application.

Table 16–4 (Cont.) Profile Configuration Options

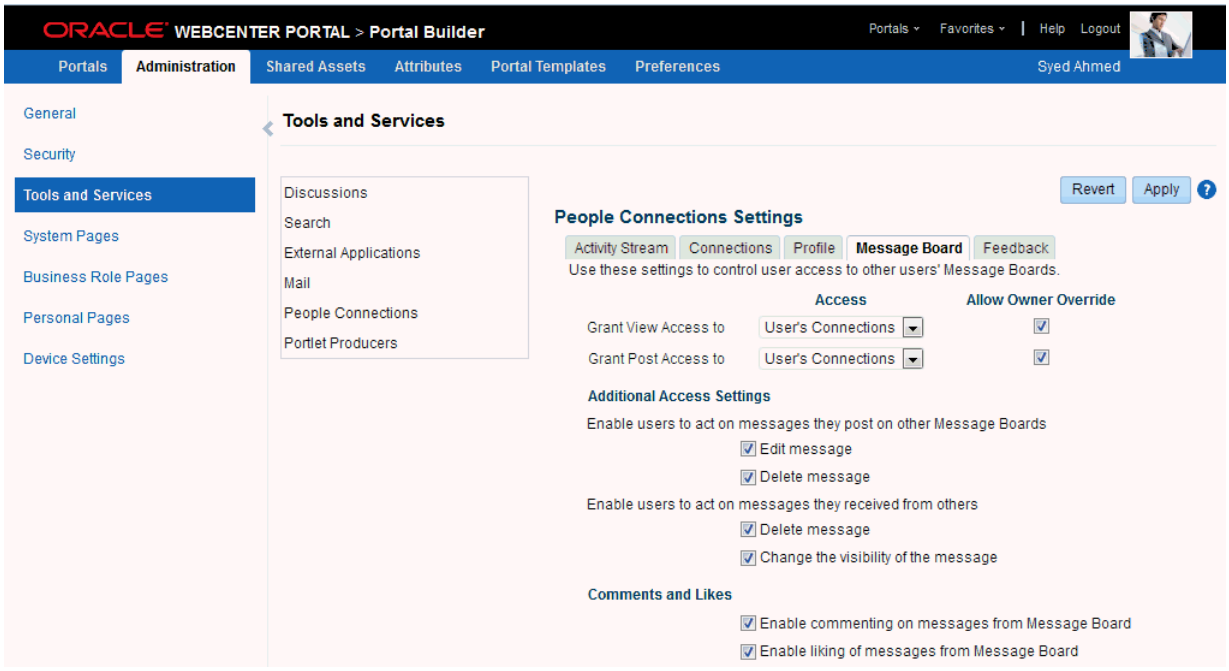
Option	Description
Profile Access	Specify which Profile sections to show and whether users are allowed to update their profile details Set application defaults in the following table columns: **Profile Section**—Identifies the groups of information shown in a user profile. **View Settings**—Specify which users can view a particular profile section, and indicate whether users can change these defaults in their personal Preferences. View Settings for the Summary section control not only who can view summary details but also for whom the user appears in people search results. Set values for: ■ **Who can view this section**—Specify which class of users can view the associated profile section by default. Choose from: **Everyone**—All users, including users who are not logged in, can see the associated profile section in other users' profiles. **Authenticated users**—Only users who are logged in can see the associated profile section in other users' profiles. **User's Connections**—The users to whom the current user is connected can see the associated profile section in other users' profiles. This option is available for all sections except **Summary**. **User Only**—Only the user can see his or her own details in the associated profile section. **None**—The section is hidden from all users. ■ **Allow Owner Override**—Enable or disable users' from overriding the default application settings you specify here. Select to enable; deselect to disable. **Can Edit**—Select to enable users to edit the associated profile section of their own personal profiles; deselect to prohibit users from editing the associated profile section. This setting also controls whether an **Edit** link appears in the Profile task flow, but it does not affect the appearance of the **Edit** button or links on the default version of the **Profile** page. You can use the other Profile administrative settings to prohibit users from actually changing any Profile details.
Profile Attributes - Edit Settings	Indicate the section attributes that users are allowed to edit by default Under **Allow Update**: ■ Select an attribute to enable users to edit its value in their own profiles. ■ Deselect an attribute to prohibit users from editing it in their own profiles.
Profile Cache settings	A means of enabling and configuring a cache for Profile details Provide values for: ■ **Number of profile objects to keep in the cache**—Enter or select a value for the size of the cache. The default is 1000. ■ **Idle time in minutes to keep a profile object in cache**—Enter or select the number of minutes to hold objects in cache before the cache is refreshed. The default is 60 minutes. ■ **LDAP read batch size for profile synchronization**—Enter or select a value for the number of LDAP profiles per batch during profile synchronization. The default is 1000. ■ **Synchronize user profile photos with LDAP when the cache expires**—Select to synchronize user profile photos periodically with LDAP (after WebCenter Portal's profile cache expires); deselect to require users to manually upload new photos through their profile page. Cache settings take effect once you restart the WebCenter Portal.

4. Click **Apply** to save your configuration settings.

16.3.5 Configuring Message Board

Message Boards provide users with a means of viewing and posting messages to their connections. Configuration settings for Message Board provide controls for who can view and post messages, who can edit and delete the messages they leave, who can delete and change the visibility of messages they receive, and whether commenting and liking are available on each message (Figure 16–4).

Figure 16–4 Configuration Settings for Message Board



See Also: For information about likes and comments, see the "Liking, Commenting On, and Sharing Items in WebCenter Portal" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

To configure Message Board:

1. Open WebCenter Portal Administration.

   For details, see Section 47.2, "Accessing the Portal Builder Administration Page"

2. Click **Tools and Services**, and then select **People Connections**.

   Alternatively, use the following URL, and then select **People Connections**:

   `http://host:port/webcenter/portal/admin/tools`

   Tabs with the names of People Connections features appear to the right.

3. Click the **Message Board** tab to bring it forward.

   Table 16–5 lists and describes each option.

Table 16–5 Message Board Configuration Options

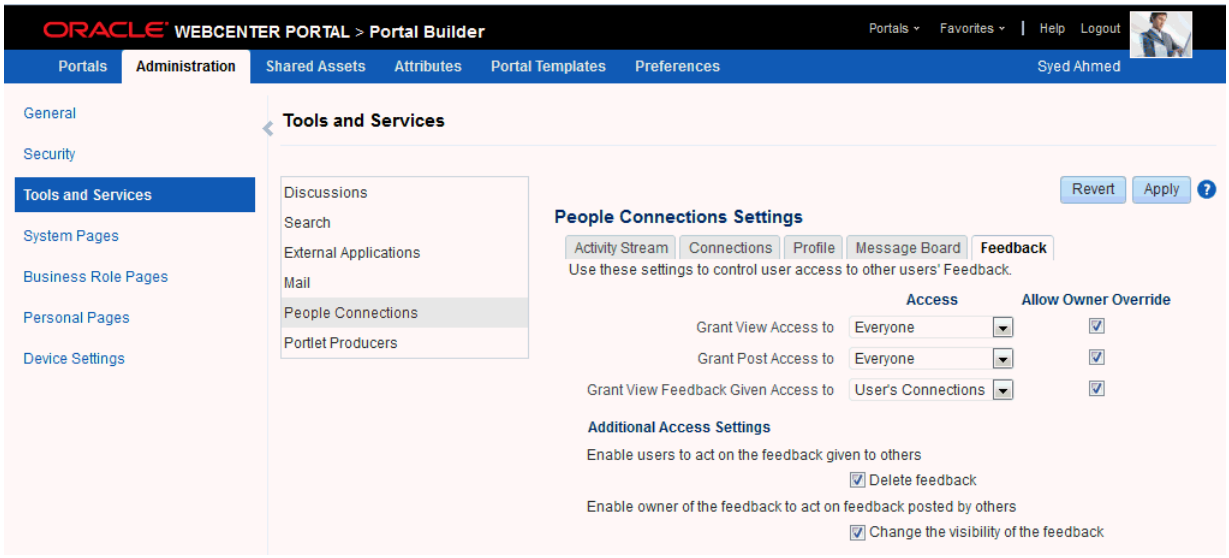
Option	Description
Grant View Access to	Specify who can view Message Board messages. ■ **Everyone**—All users, whether logged in or not, can see users' Message Board messages. ■ **Authenticated Users**—Only logged in users can see users' Message Board messages. ■ **User's Connections**—Only the user and the user's connections can view the user's Message Board. ■ **User Only**—Only the user can see the messages on his or her Message Board.
Grant Post Access to	Specify who can post Message Board Messages. ■ **Everyone**—All users, whether logged in or not, can post Message Board messages. ■ **Authenticated Users**—Only logged in users can post messages to Message Boards. ■ **User's Connections**—Only the user and the user's connections can post messages to the user's Message Board. ■ **User Only**—Only the user can post messages to his or her Message Board.
Allow Owner Override	Specify whether users can override these administrative defaults. ■ Select to enable users to edit the default settings through user Preferences (for more information, see the "Setting People Connections Preferences" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*). ■ Deselect to enforce the administrator default application settings.
Enable users to act on messages they post on other Message Boards	Specify whether users are allowed to act on the messages they post. ■ **Edit message**—Select to enable users to edit their own Message Board posts; deselect to prohibit users from editing the messages they post. ■ **Delete message**—Select to enable users to delete their own Message Board posts; deselect to prohibit users from deleting the messages they post.
Enable users to act on messages they received from others	Specify whether users can act on messages they receive from others ■ **Delete message**—Select to enable users to delete messages they receive from other users; deselect to prohibit users from deleting the messages they receive. ■ **Change the visibility of the message**—Select to enable users to hide or show the messages from a given user; deselect to prohibit users from hiding or showing messages.
Enable commenting on messages from Message Board	Specify whether users can comment on messages that are posted on a Message Board. ■ Select to permit users to comment on messages. A **Comment** link appears below each message. Users click this to enter a comment. For more information, see the "Liking, Commenting On, and Sharing Items in WebCenter Portal" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*. ■ Deselect to prohibit commenting.
Enable liking of messages from Message Board	Specify whether to enable users to indicate that they like a message. ■ Select to permit users to like messages. A **Like** link appears below each message. For more information, see the "Liking, Commenting On, and Sharing Items in WebCenter Portal" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*. ■ Deselect to prohibit liking.

4. Click **Apply** to save your configuration settings.

16.3.6 Configuring Feedback

Feedback provides a means of viewing and posting user feedback for other application users. Configuration settings for Feedback (Figure 16–5).

Figure 16–5 Configuration Settings for Feedback



Feedback configuration settings offer controls for identifying who can view, post, and delete feedback.

To configure Feedback:

1. Open WebCenter Portal Administration.

   For details, see Section 47.2, "Accessing the Portal Builder Administration Page"

2. Click **Tools and Services**, and then select **People Connections**.

   Alternatively, use the following URL, and then select **People Connections**:

   `http://host:port/webcenter/portal/admin/tools`

   Tabs with the names of People Connections features appear to the right.

3. Click the **Feedback** tab to bring it forward.

   Table 16–6 lists and describes each option.

Table 16–6 Feedback Configuration Options

Option	Description
Grant View Access to	Specifies who can view the current user's Feedback ■ **Everyone**—All users, whether logged in or not, can see a given user's Feedback. ■ **Authenticated Users**—Only users who are logged in can see a given user's Feedback. ■ **User's Connections**—Only the user and the user's connections can see a given user's Feedback. ■ **User Only**—Disables other users from viewing a given user's Feedback.
Grant Post Access to	Specifies who can post user Feedback ■ **Everyone**—All users, whether logged in or not, can post Feedback for a given user. ■ **Authenticated Users**—Only logged in users can post Feedback for a given user. ■ **User's Connections**—Only the user and the user's connections can post Feedback for a given user. ■ **User Only**—Users can post Feedback only for themselves. Effectively disables Feedback.
Grant View Feedback Given Access to	Specifies who can see the **View** menu to switch between Feedback Given and Feedback Received in a Feedback task flow ■ **Everyone**—All users, whether logged in or not, can see the options on the **View** menu. ■ **Authenticated Users**—Only logged in users can see the options on the **View** menu. ■ **User's Connections**—Only the user and the user's connections can see the **View** menu. ■ **User Only**—Disables the View menu for all but the current user. When users visit the current user's Feedback task flow, they can view only the Feedback the current user has received.
Allow Owner Override	Specifies whether users can override these administrative defaults ■ Select to enable users to revise application default settings through user preferences. (For more information, see the "Setting Feedback Preferences" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.) ■ Deselect to prevent users from altering administrator settings for Feedback.
Enable users to act on the feedback given to others	Indicates whether users can delete the Feedback they post ■ Select **Delete feedback** to enable users to delete the Feedback they post. ■ Deselect **Delete feedback** to prohibit users from deleting the Feedback they post.
Enable owner of the feedback to act on feedback posted by others	Indicate whether to enable users to hide or show Feedback from another user. ■ Select **Change the visibility of the feedback** to enable users to hide or show the Feedback from another user. ■ Deselect **Change the visibility of the feedback** to prohibit users from hiding or showing Feedback left by others.

4. Click **Apply** to save your configuration settings.

16.4 Setting Up a Proxy Server for Activity Stream

To enable external people connection feeds in WebCenter Portal or your Portal Framework application, you must set up a proxy server.

A proxy server is also required if you want to display external links in Activity Stream task flows. Both the People Connections service and the Activity Stream service share the same proxy server settings.

You can configure a proxy server by using either Fusion Middleware Control or WLST. For information, see Section 8.2.4, "Setting Up a Proxy Server."

16.5 Archiving the Activity Stream Schema

Administrators can use WLST commands to archive and restore data in the Activity Stream schema. The following commands are available:

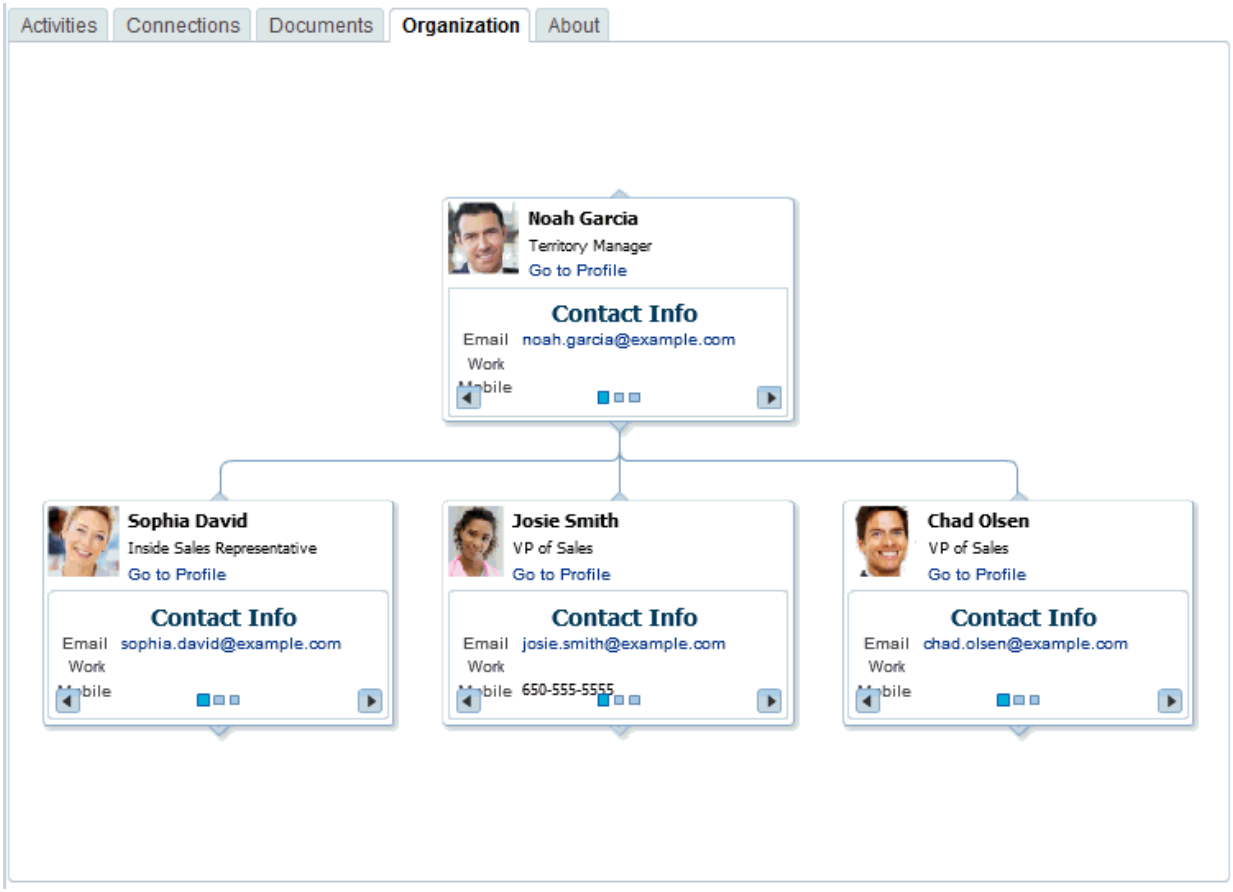
- `archiveASByDate`—Archive activity stream data that is older than a specified date.
- `archiveASByDeletedObjects`—Archive activity stream data associated with deleted objects.
- `archiveASByClosedSpaces`—Archive activity stream data associated with portals that are currently closed.
- `archiveASByInactiveSpaces`—Archive activity stream data associated with portals that have been inactive since a specified date.
- `restoreASByDate`—Restore archived activity stream data from a specified date into production tables.
- `truncateASArchive`—Truncates activity stream archive data.
- `archiveASBySpace`—Archive activity stream data associated with a portal.
- `archiveASAllSpaces`—Archive activity stream data associated with all portals.
- `archiveASByUser`—Archive activity stream data associated with a user.
- `archiveASAllUsers`—Archive activity stream data associated with all users.
- `archiveASByDeletedActors`—Archive activity stream data associated with deleted actors.
- `showASStatistics`—Report activity stream statistics.

For more information, see the "Activity Stream" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

16.6 Specifying a Management Chain for Organization View

The Organization View task flow and the **Organization** tab on a **Profile** page can provide a visualization of your management chain, that is, they can render a view of a manager and the manager's direct reports (Figure 16–6).

Figure 16–6 Organization View of a Manager and the Manager's Direct Reports



By default, the values that define the management chain for these organization views are blank. This means that managers are not automatically specified for users in the back-end identity store that provides user details.

> **Tip:** The value for **Manager** on the **Profile** page's **About** tab is also defined by the methods suggested in this section.

For the management chain to be rendered in organization views, the back-end identity store that is used for WebCenter Portal authentication must be set up in such a way that direct report users have a `manager` attribute. And the `manager` attribute must be defined as the Distinguished Name (DN) of their manager user (see "Example Embedded LDAP Configuration").

> **Tip:** In an LDAP environment, a user can be managed by only one person; in the same environment, a user can manage many people.

Example Embedded LDAP Configuration

You can specify a management chain within the Oracle WebLogic Server (WLS) embedded LDAP or within an external LDAP, such as Oracle Internet Directory (OID). However, the management chain you define through the embedded LDAP is for testing or proof of concept and not for production. For production, you must use an external LDAP, such as OID, for the identity store for WebCenter Portal authentication.

See Also: For more information, see Chapter 31, "Configuring the Identity Store," or refer to the documentation provided with your LDAP implementation.

This example describes how to define a management chain within the embedded LDAP in WebLogic Server for testing or proof of concept.

Note: The steps provided in this example are similar to those you take for an external LDAP. That is, you create an attribute (`manager`) and set a value on the attribute for each user. For this value, enter the DN of the selected user's manager.

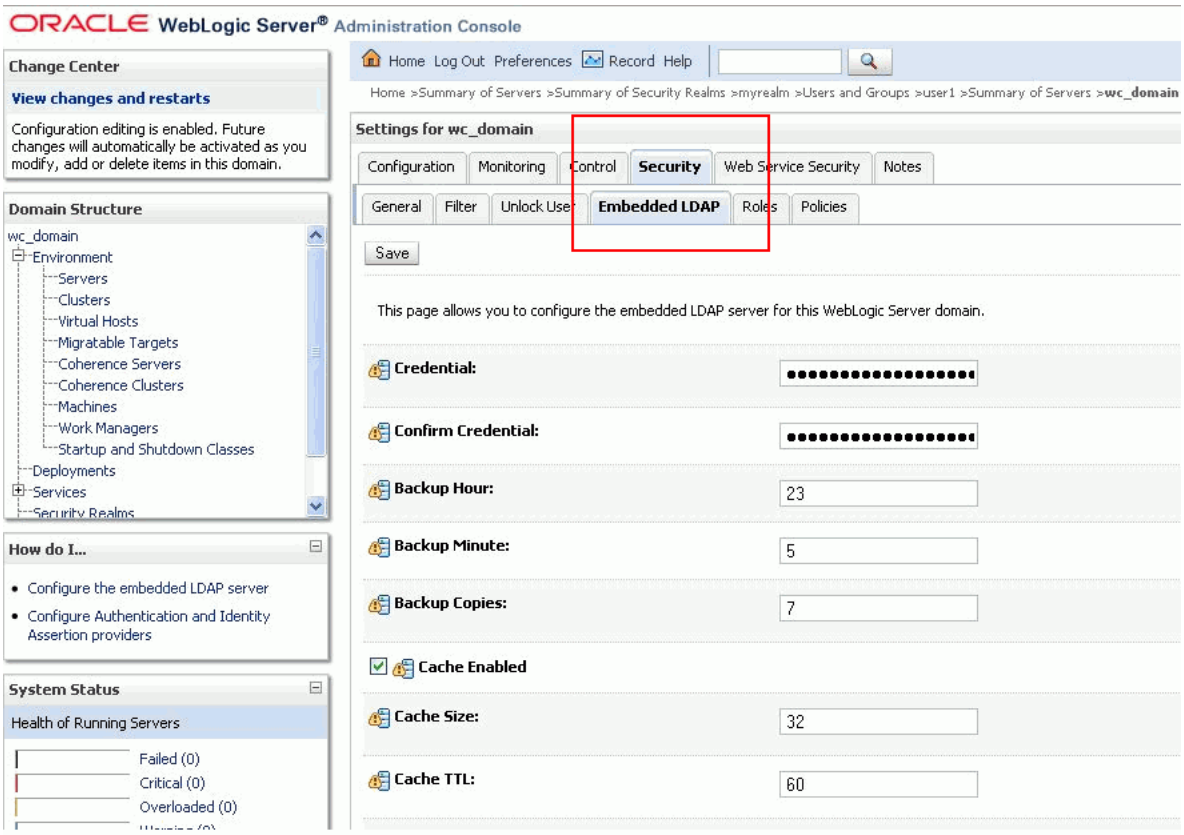
In this example, there are three users:

- `user1`
- `user2`
- `manager_user`

To define a management chain with these users:

1. Enable browsing of the embedded LDAP using an external viewer, such as Apache Directory Studio:
   a. Go to the WLS Administration Console, and log in as the administrator user.
   b. Click your domain, for example, wc_domain, then open the Security Tab and then the Embedded LDAP Sub-tab (Figure 16–7).

Figure 16–7 Oracle WebLogic Server Administration Console



c. Enter a value in the **Credential** field, and then reenter that value in the **Confirm Credential** field.

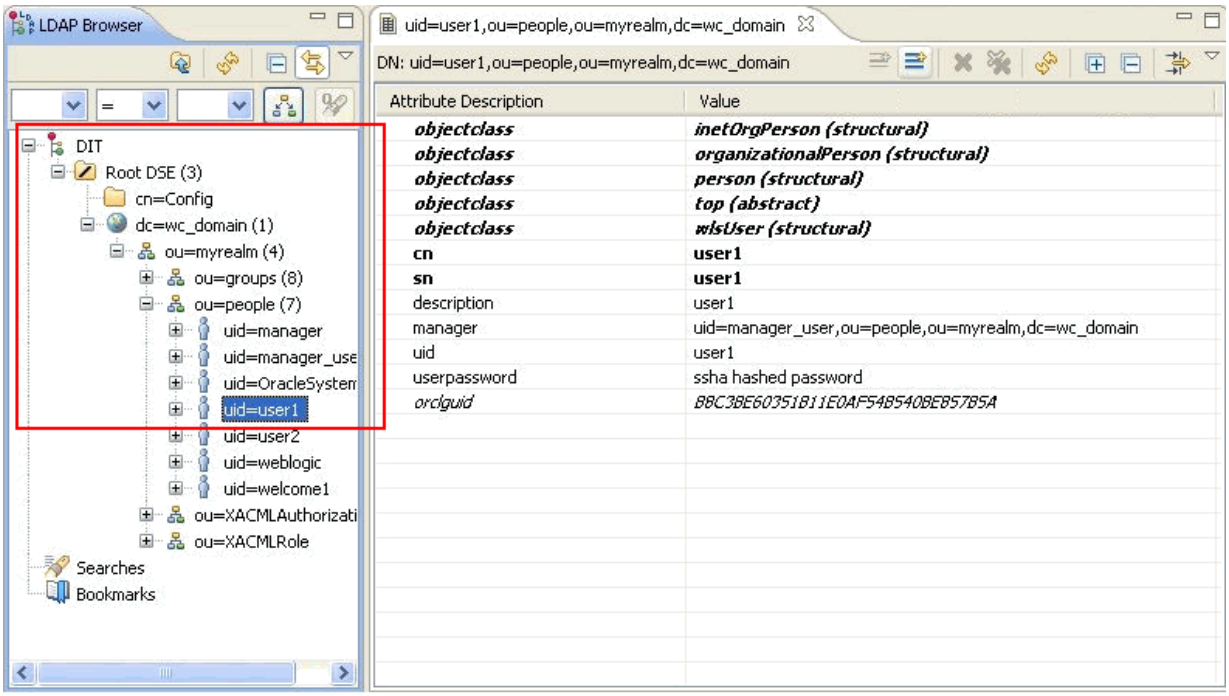
Tip: The default credential is a randomly generated password. Set it to something memorable.

d. Restart your administration and managed servers.

2. Start up the LDAP viewer you selected in Step 1, and create a connection using the following details:
   - `hostname` (for example, `example.com`)
   - `port` (the WLS administration port, for example `7001`)
   - `Bind DN` (`cn=Admin`)
   - `Password` (that is, the credential you set in Step 1c)
3. Navigate to `user1` by finding the users within the DIT tree (Figure 16–8).

   For example, click in succession:
   - `dc=wc_domain`
   - `ou=myrealm`
   - `ou=people`
   - `uid=user1`

Figure 16–8 Selecting a User in the DIT Tree of an LDAP Browser



4. In the **Attribute Description** column, add a new attribute of type `manager`.

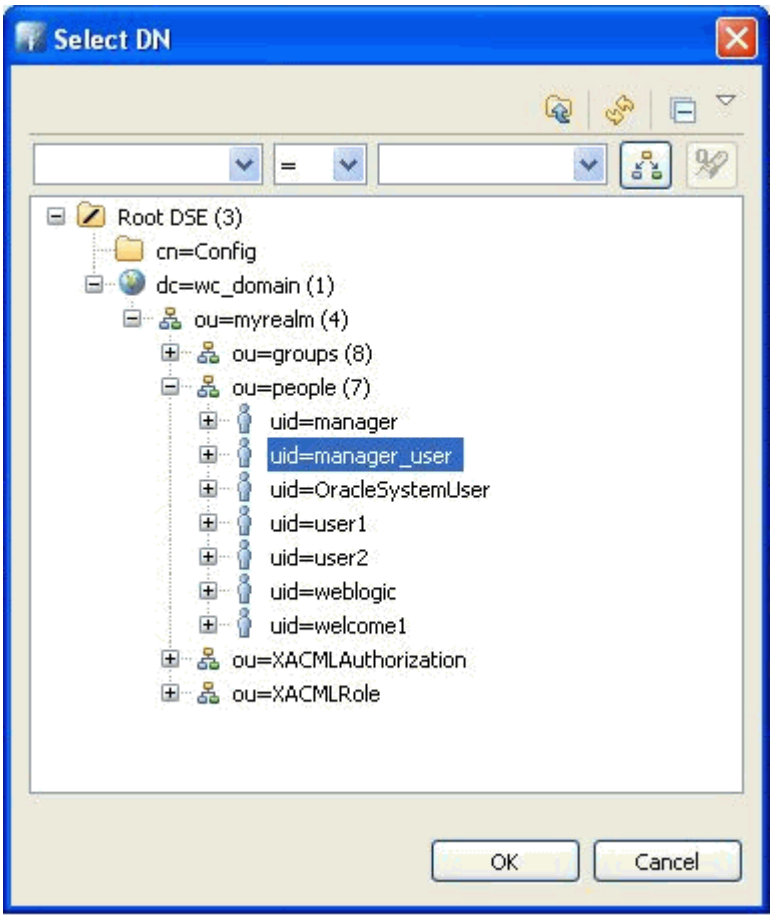
   Tip: Press Ctrl-Shift-+ to open the New Attribute dialog.

5. For the attribute value, select the DN for `manager_user` (Figure 16–9).

   For example, under the root, select in succession:

   - `dc=wc_domain`
   - `ou=myrealm`
   - `ou=people`
   - `uid=manager_user`

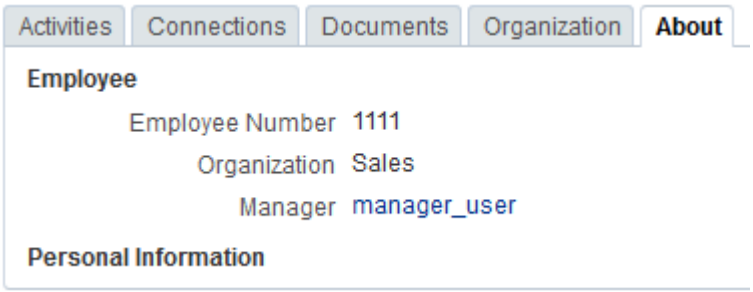
Figure 16–9 Select DN Dialog



6. Repeat Steps 3 through 5 for `user2`.

Now `user1` and `user2` are managed by `manager_user`. You can check this by logging in to WebCenter Portal as `user1` and navigating to the **About** tab of the **Profile** page. The user `manager_user` is displayed as the manager (Figure 16–10).

Figure 16–10 About Tab of the Profile Page



Tip: Click the value for **Manager** (in this example, `manager_user`) to view the manager's profile. Access the **Organization** tab to see the organization view associated with the currently viewed profile.

16.7 Setting Profile Configuration Properties

Administrators can use WLST commands to set profile configuration properties, such as setting the profile version that appears in the user interface. Administrators can perform the following actions:

- Set the profile configuration properties by running `setProfileConfig`.

  Syntax:

  ```
  setProfileConfigProperties(appName,[ProfilePageVersion],
  [ProfileCacheNumberOfObjects],
  [ProfileCacheTimeToLive],[ProfileSyncLDAPReadBatchSize],
  [ProfileSyncHourOfDay], [ProfileSyncFrequencyInDays],
  [server],[applicationVersion])
  ```

  This command takes the following parameters:

  – `appName` - The name of the WebCenter Portal application in which to perform this operation. For example, `webcenter`.

  – `ProfilePageVersion` - (Optional) The profile page version to use. Valid values for ProfilePageVersion are:

    * `v1` - Use old-style Profile pages (11.1.1.7.0 and earlier)
    * `v2` - (default) Use the new Profile page format (introduced in 11.1.1.8.0)

    Note: Profile page version changes will not take effect until you restart the server on which the WebCenter Portal application is deployed.

  – `ProfileCacheNumberOfObjects` - (Optional) The number of profile objects to keep in the profile cache. Any value between 1 and 10000. The default value is 1000.

  – `ProfileCacheTimeToLive` - (Optional) The length of time (in minutes) to keep a profile object in the cache. Any value between 1 and 1440. The default value is 60.

  – `ProfileSyncLDAPReadBatchSize` - (Optional) The LDAP read batch size that is used during profile synchronization. Any value between 1 and 1000. The default value is 1000.

  – `ProfileSyncHourOfDay` - (Optional) When (the hour) to start profile synchronization. Any value between 0 and 23. The default value is 23, equivalent to 11pm.

  – `ProfileSyncFrequencyInDays` - (Optional) How often profile synchronization takes place (in days). Any value greater than 0. The default value is 7.

    Note: If you omit a parameter, the corresponding configuration remains unchanged.

- List the current profile configuration settings by running `listProfileConfig`.

  Syntax:

  ```
  listProfileConfig(appName)
  ```

This command takes the following parameter:

- `appName` - The name of the WebCenter Portal application in which to perform this operation. For example, `webcenter`.

- Get the current value of a profile property by running `getProfileConfig`.

  Syntax:

  ```
  getProfileConfig(appName, key, [server], [applicationVersion])
  ```

  This command takes the following parameters:

  - `appName` - The name of the WebCenter Portal application in which to perform this operation. For example, `webcenter`.
  - `key` - Name of a the Profile Config property to get. Valid values include:
    * `ProfilePageVersion`
    * `ProfileCacheNumberofObjects`
    * `ProfileCacheTimeToLive`
    * `ProfileSyncLDAPReadBatchSize`
    * `ProfileSyncHourOfDay`
    * `ProfileSyncFrequencyInDays`
  - `server` - (Optional) The name of the target server on which the application is deployed.
  - `applicationVersion` - (Optional) The version number of the application.

16.8 Synchronizing Profiles with the Identity Store

Administrators can use WLST commands to synchronize profile information in the LDAP identity store with WebCenter Portal. Administrators can perform the following actions:

- Start or stop profile synchronization for all users or a single user by running `startSyncProfiles` or `stopSyncProfiles`.
- Check whether profile synchronization is currently in progress by running `isSyncProfilesRunning`.
- Set various profile synchronization options:
  - Specify whether to synchronize user profile photos in LDAP by running `setProfilePhotoSyncEnabled`.
  - Synchronize profile information for a specific user by running `syncProfile`.

For more information, see the "WebCenter Portal Identity Store" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

Administrators can also use the WebCenter Portal user interface to control the LDAP batch size and turn on or off photo synchronization:

1. Open WebCenter Portal Administration.

   For details, see Section 47.2, "Accessing the Portal Builder Administration Page"

2. Click **Tools and Services**, and then select **People Connections**.

   Alternatively, use the following URL, and then select **People Connections**:

```
http://host:port/webcenter/portal/admin/tools
```

Tabs with the names of People Connections features appear to the right.

3. On the right, click the **Profile** tab.
4. To change the LDAP batch setting, under Profile Cache settings, set the **LDAP read batch size for profile sync task** as desired.
5. To turn on or off photo synchronization, select or clear the box next to **Synchronize user profile photos with LDAP when the cache expires**.

16.9 Configuring Cache Options for the Profile Service

The Profile service caches objects to save processing bandwidth, presenting information more quickly to users. The following are the default cache options for the Profile service:

- Maximum number of objects in the cache: 1000

  If the maximum number of objects in the cache is reached, older objects or unused objects will be removed from the cache as additional objects are added.

- Maximum idle time in cache: 60 seconds

  If an object in the cache has not been accessed for the maximum idle time it will be removed from the cache.

If you find that the profile service is performing more slowly than you want, you might want to increase the thresholds in the cache settings. If you feel that the data presented by the profile service is not updating frequently enough, you might want to decrease the cache thresholds.

Administrators can configure the Profile service cache options through the WebCenter Portal user interface or with WLST commands:

- To set cache options through the WebCenter Portal user interface:

  1. Open WebCenter Portal Administration.

     For details, see Section 47.2, "Accessing the Portal Builder Administration Page"

  2. Click **Tools and Services**, and then select **People Connections**.

     Alternatively, use the following URL, and then select **People Connections**:

     ```
     http://host:port/webcenter/portal/admin/tools
     ```

     Tabs with the names of People Connections features appear to the right.

  3. On the right, click the **Profile** tab.
  4. Under Profile Cache settings, set the **Number of profile objects to keep in cache** and **Length of time to keep a profile object in cache (in minutes)** settings as desired.

- To set cache options through WLST, use the `setProfileConfig` command, described in Section 16.7, "Setting Profile Configuration Properties."

  For command syntax and examples, see the "WebCenter Portal Identity Store" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

  For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

16.10 Troubleshooting Issues with People Connections

This section provides information to assist you in troubleshooting problems you may encounter while using People Connections.

Problem

Profile photos added to WebCenter Portal using the JDeveloper workspace `SampleWebCenterSpacesExtensions.jws` do not display

Solution

If you used WebCenter Portal's `SampleWebCenterSpacesExtensions.jws` workspace to add profile photos, but old profile photos continue to be shown, complete the following steps to delete the current photo data from the database:

1. Determine the GUID associated with the user using the following EL expression in the portal:

   ```
   #{webCenterProfile[securityContext.userName].guid}
   ```

2. Using SQLPlus, log in to the WebCenter Portal database as DBA (or another administrative user), and execute the following commands:

   ```
   DELETE FROM WcPeopleConnProfilePhoto photo WHERE photo.userGuid =USER_GUID;
   ```

   Replace *`USER_GUID`* with the user GUID you recorded in step 1.

17 Managing RSS

This chapter describes how to configure and manage RSS functionality for WebCenter Portal and Portal Framework applications.

This chapter includes the following topics:

- Section 17.1, "About RSS"
- Section 17.2, "RSS Prerequisites"
- Section 17.3, "Setting Up a Proxy Server for External RSS News Feeds"
- Section 17.4, "Testing External RSS News Feed Connections"

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console and the `Administrator` role in the deployed application:

- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.
- **Portal Framework application**: `Administrator` role granted through the Administration Console.

For more information about roles and permissions, see Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

17.1 About RSS

The RSS functionality encompasses a RSS Viewer and RSS service that shows news feeds from various WebCenter Portal tools and services. The RSS Viewer enables users to view external news feeds from different web sites inside WebCenter Portal and Portal Framework applications. RSS also delivers content update information from various portal resources including recent activities, discussions, lists, and announcements.

17.2 RSS Prerequisites

RSS functionality does not require any back-end server. You do not need to set up a connection to use it. However, depending on your network configuration, you may need to set up a proxy server to enable WebCenter Portal or your Portal Framework application to display content from external RSS news feeds.

17.3 Setting Up a Proxy Server for External RSS News Feeds

To enable external RSS news feeds in WebCenter Portal or a Portal Framework application, you must set up a proxy server.

A proxy server is also required if you want to display external links in Activity Stream task flows. Both RSS and the activity stream share the same proxy server settings.

You can configure a proxy server by using either Fusion Middleware Control or WLST. For information, see Section 8.2.4, "Setting Up a Proxy Server."

17.4 Testing External RSS News Feed Connections

After setting up the proxy server for the RSS Viewer, you can test the connection to make sure you can access external RSS feeds.

To ensure the proxy server is accurately configured for the RSS Viewer:

1. In WebCenter Portal or your Portal Framework application, add the RSS task flow to a page.

   For information about adding the RSS task flow and editing the URL, see the "Adding RSS News Feeds to a Portal" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Edit the RSS task flow and set the URL to an external RSS feed.

   For example:

   ```
   http://www.oracle.com/ocom/groups/public/@ocom/documents/webcontent/196280.xml
   ```

   If the RSS feed displays correctly, proxy configuration is set up properly.

18

Managing Oracle Secure Enterprise Search in WebCenter Portal

This chapter describes how to configure Oracle Secure Enterprise Search (SES) release 11.2.2.2 to index and search most objects in both WebCenter Portal and Portal Framework applications.

This chapter includes the following topics:

Tip: Oracle WebCenter Portal provides a script that can help configure Oracle SES for evaluation purposes. After you read and consider the configuration described in this chapter, see Appendix D, "Oracle Secure Enterprise Search Configuration for Evaluation" for another configuration option.

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console and the `Administrator` role in the deployed application:

- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.
- **Portal Framework application**: `Administrator` role granted through the Administration Console.

For more information about roles and permissions, see Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

18.1 About Search with Oracle SES

Oracle recommends Oracle SES for best search performance. In addition to providing better scalability and performance than Oracle WebCenter Portal's own live (delegated) search, Oracle SES is beneficial for the following reasons:

- Oracle SES provides unified ranking results, with the most relevant items appearing first.
- Oracle SES provides more thorough search. For example, when searching lists in WebCenter Portal, the live search only searches list names and descriptions, while Oracle SES also searches list column names and column contents.
- Oracle SES allows search of other repositories outside of WebCenter Portal or your Portal Framework application.
- Oracle SES supports the search REST APIs and data control for customizing your search interface.

> **Note:** The steps in this chapter are *not* required if you choose to use Oracle WebCenter Portal's live search adapter.

Your search environment varies depending on the version of Oracle SES configured. For example:

- If you configure your system with **Oracle SES 11.2.2.2**, then you can customize search in the administration settings for the application. Oracle SES 11.2.2.2 supports faceted search, filtered search in the search box, and document thumbnails, while earlier releases of Oracle SES and implementations with live (delegated) search do not. All customization with Oracle SES 11.2.2.2 is done on the **Tools and Services - Search** administration page, even though task flow parameters may display.

  To access the Search administration page, on the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), select **Tools and Services**, then **Search**.

  > **Note:** This task is performed by an administrator. Working with search at the portal level is an application specialist or portal moderator task, as described in the "Adding Search to a Portal" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

- If you configure your system with **Oracle SES 11.1.2.***, then you can customize search using Search (Search - Non-Faceted Search) task flow parameters. Oracle SES 11.1.2.* supports saving searches and setting user preferences with search, while the 11.2.2.2 adapter and implementations with live (delegated) search do not.

  > **Note:** This chapter describes how to configure WebCenter Portal with Oracle SES release 11.2.2.2. If you choose to use an earlier release of Oracle SES, see:
  >
  > `http://docs.oracle.com/cd/E28280_01/webcenter.1111/e10148/jpsdg_search.htm#BABBDFDC`

With WebCenter Portal applications, live (delegated) search is set as the default search platform; however, large-scale implementations should be configured to use Oracle SES for best performance. (Detailed configuration steps are in Section 18.5, "Configuring Oracle SES to Search WebCenter Portal Applications.")

With **Portal Framework applications**, Oracle SES is set as the default and preferred search platform. (Detailed configuration steps are in Section 18.6, "Configuring Oracle SES to Search Portal Framework Applications.")

You can configure Oracle SES to search the following resources:

- Documents, including wikis and blogs
- Announcements and discussions
- Portals, lists, page metadata, and people

> **Note:** The crawler that indexes portals, lists, page metadata, and people is not supported in Portal Framework applications, so those resources are not searchable with Oracle SES in Portal Framework applications.
>
> When Oracle SES is configured to search WebCenter Portal or Portal Framework applications, other non-crawled resources (for example, notes and events) are not returned in search results.

Results from all sources are listed together, with the most relevant items appearing first. For example, when you run a search for a user name, most likely, you are looking for that person's contact information (that is, the exact user name in People Connections), not necessarily documents that the user wrote.

Three types of Oracle SES crawlers index WebCenter Portal resources:

- **Documents Crawler**: This uses the Oracle SES Oracle Content Server source type to crawl documents, including wikis and blogs.
- **Discussions Crawler**: This uses the Oracle SES Database source type to crawl discussions and announcements.
- **Spaces Crawler**: (Not available in Portal Framework applications.) This uses the Oracle SES Oracle WebCenter source type to crawl certain objects, such as lists, page metadata, portals, and profiles.

> **Note:** Oracle SES crawlers collect data through connectors to back-end repositories. Each connector is configured in Oracle SES as a "crawl source." Each crawl source has a type that identifies the type of repository that holds the data, such as a relational database or an Oracle Content repository.

When you configure Oracle SES, all available Oracle SES crawlers should be enabled. It is not recommended to enable one Oracle SES crawler and not another. For example, when you configure Oracle SES to search Portal Framework applications, you cannot have it crawl documents but not discussions.

18.2 Configuration Roadmaps for Oracle SES in WebCenter Portal

Use the roadmaps in this section as an administrator's guide through the configuration process:

- **Roadmap - Configuring Oracle SES**

  Figure 18–1 and Table 18–1 provide an overview of the prerequisites and tasks required to get Oracle SES working.

Figure 18–1 Configuring Oracle SES

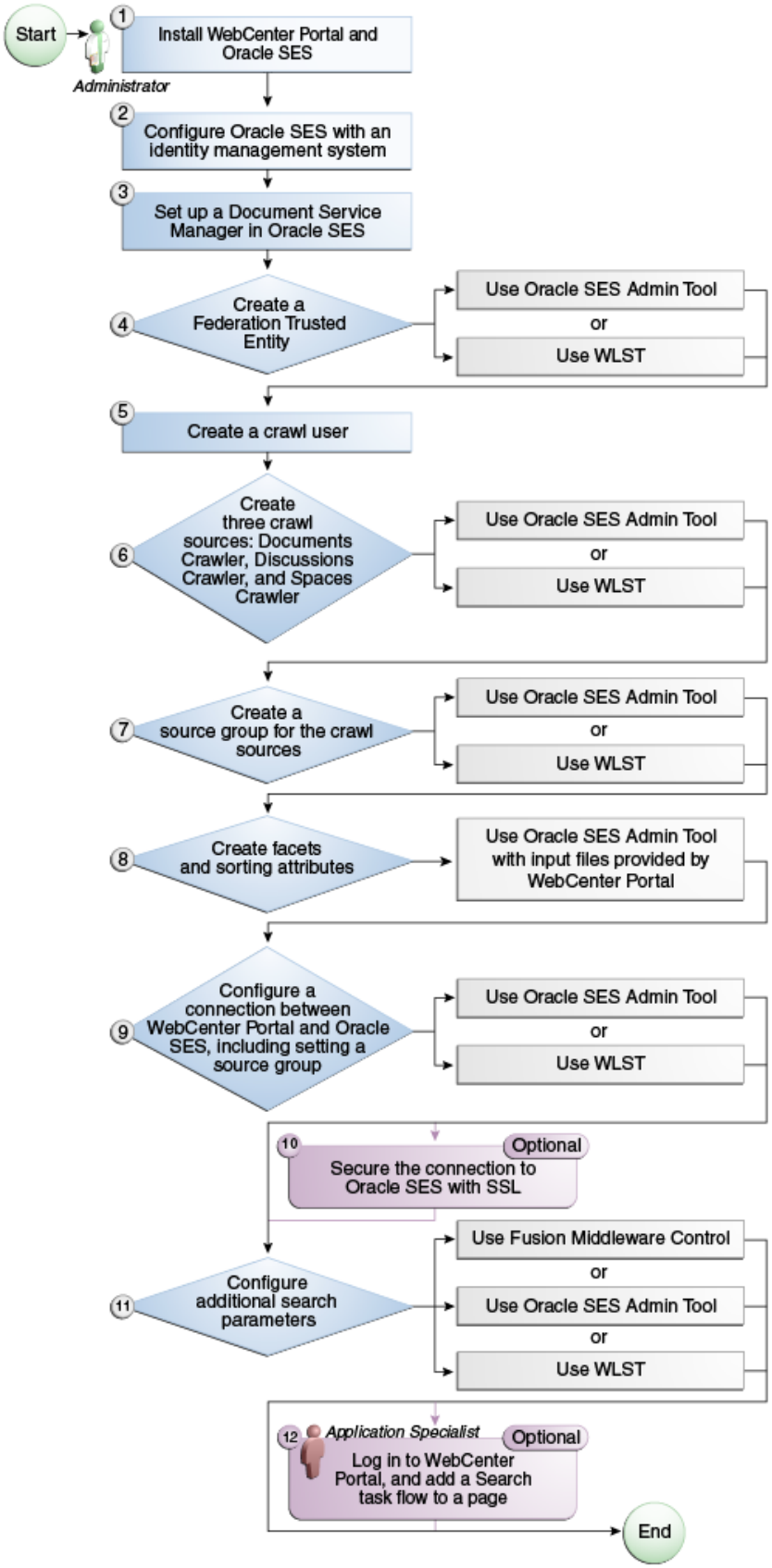


Table 18–1 Configuring Oracle SES

Actor	Task
Administrator	1. Install WebCenter Portal and Oracle SES
	2. Configure Oracle SES with an identity management system
	3. Set up a Document Manager in Oracle SES
	4. Create a Federation Trusted Entity using one of the following tools: ■ Oracle SES Admin Tool ■ WLST
	5. Create a crawl user
	6. Create three crawl sources: documents crawler, discussions crawler, and spaces crawler using one of the following tools: ■ Oracle SES Admin Tool ■ WLST
	7. Create a source group for the crawl sources using one of the following tools: ■ Oracle SES Admin Tool ■ WLST
	8. Create facets and sorting attributes
	9. Configure a connection between WebCenter Portal and Oracle SES using one of the following tools: ■ Oracle SES Admin Tool ■ WLST **Note**: This step must include running the `setSESVersion` WLST command to enable the Tools and Services - Search administration page.
	10. (Optional) Secure the connection to Oracle SES with SSL
	11. Configure additional search parameters using one of the following tools: ■ Fusion Middleware Control ■ Oracle SES Admin Tool ■ WLST
Application Specialist	12. (Optional) Add a search task flow to a portal

- **Roadmap - Configuring Oracle SES for Portal Framework Applications**

  Figure 18–2 and Table 18–2 provide an overview of the prerequisites and tasks required to get Oracle SES working in Portal Framework applications.

Figure 18–2 Configuring Oracle SES for Portal Framework Applications

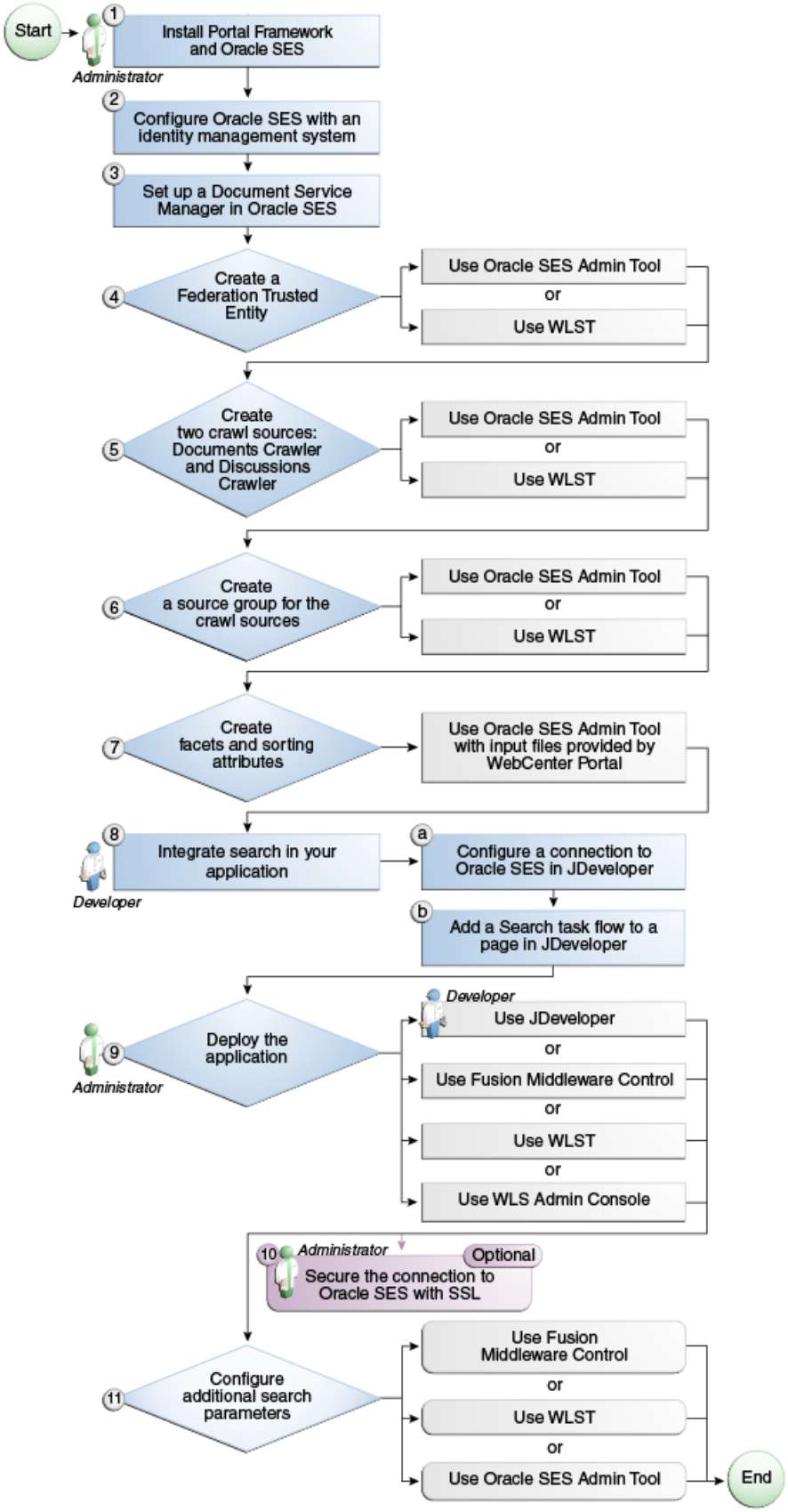


Table 18–2 Configuring Oracle SES for Portal Framework Applications

Actor	Task	Sub-task
Administrator	1. Install WebCenter Portal and Oracle SES	
	2. Configure Oracle SES with an identity management system	
	3. Set up a Document Service Manager in Oracle SES	
	4. Create a Federation Trusted Entity using one of the following tools: ■ Oracle SES Admin Tool ■ WLST	
	5. Create two crawl sources: documents crawler and discussions crawler using one of the following tools: ■ Oracle SES Admin Tool ■ WLST	
	6. Create a source group for the crawl sources using one of the following tools: ■ Oracle SES Admin Tool ■ WLST	
	7. Create facets and sorting attributes	
Developer	8. Integrate search in your Portal Framework application	8.a Configure a connection to Oracle SES in JDeveloper **Note**: This step must include running the `setSESVersion` WLST command to enable the Tools and Services - Search administration page.
		8.b Add a search task flow to a page in JDeveloper
Developer or Administrator	9. Deploy the Portal Framework application using one of the following tools: ■ JDeveloper (Developer) ■ Fusion Middleware Control (Administrator) ■ WLST (Administrator) ■ WLS Admin Console (Administrator)	
Administrator	10. (Optional) Secure the connection to Oracle SES with SSL	
Administrator	11. Configure additional search parameters using one of the following tools: ■ Fusion Middleware Control ■ WLST ■ WLS Admin Console	

18.3 Prerequisites for using Oracle SES

This section includes the following topics:

- Section 18.3.1, "Oracle SES - Installation"
- Section 18.3.2, "Oracle SES - Configuration"
- Section 18.3.3, "Oracle SES - Security"

18.3.1 Oracle SES - Installation

This chapter assumes that you are running the latest supported products: Oracle SES 11.2.2.2 and Oracle WebCenter Content Server 11.1.1.8.

Supported Oracle SES versions include 11.2.2.2, 11.1.2, and 11.1.2.2. Oracle recommends using Oracle SES release 11.2.2.2 for best performance and the latest search features.

Note: This chapter describes how to configure WebCenter Portal with Oracle SES release 11.2.2.2. If you choose to use an earlier release of Oracle SES, see:

`http://docs.oracle.com/cd/E28280_01/webcenter.1111/e10148/jpsdg_search.htm#BABBDFDC`

In addition to the Oracle SES installation, WebCenter Portal requires some post installation steps. For complete instructions, see the "Back-End Requirements for Search" section in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*.

See Also: It is important to verify that you have installed all required patches for Oracle SES. For the latest information on required patches, check the Release Notes.

18.3.2 Oracle SES - Configuration

1. Oracle SES must be configured with an identity management system to validate and authenticate users. This is necessary for secure searches, so searches return only results that the user is allowed to view based on access privileges.

   Because WebCenter Portal uses identity propagation when communicating with Oracle SES, WebCenter Portal's user base must match that in Oracle SES. One way this can happen is by configuring WebCenter Portal and Oracle SES to the same identity management system, such as Oracle Internet Directory.

   Note: For information on all supported identity management systems, see Section 30.2.3, "Default Identity and Policy Stores."

   Only one identity plug-in can be set up for each Oracle SES instance. All repositories (Oracle WebCenter Content Server, Oracle WebCenter Portal Discussions Server, and Oracle WebCenter Portal) must share the same user base as Oracle SES.

   Oracle SES includes numerous identity plug-ins for identity management systems including Oracle Internet Directory, Oracle WebCenter Content Server, and Microsoft Active Directory. For information, see the Oracle SES documentation included with the product. (This is listed on the WebCenter Portal product area on the Oracle Fusion Middleware documentation library.)

The following example sets up the identity plug-in for Oracle Internet Directory:

a. In the Oracle SES administration tool, navigate to the Global Settings - Identity Management Setup page, select **Oracle Internet Directory** from the available identity plug-ins, and click **Activate**.

b. Provide the following values:

   Host name: The host name of the computer where Oracle Internet Directory is running

   Port: The Oracle Internet Directory port number

   Use SSL: `true` or `false`

   Realm: The Oracle Internet Directory realm, for example, `dc=us,dc=oracle,dc=com`

   User name: The Oracle Internet Directory admin user name; for example, `cn=orcladmin`

   Password: Admin user password

c. Click **Submit**.

2. Each Oracle SES instance must have a trusted entity for allowing WebCenter Portal end users to be securely propagated at search time. (A trusted entity allows the WebCenter Portal to authenticate itself to Oracle SES and assert its users when making queries on Oracle SES.) This trusted entity can be any user that either exists on the identity management server behind Oracle SES or is created internally in Oracle SES.

   You can do this either in WLST or in Oracle SES.

   Note: This trusted entity name and password is required later as the `appUser` and `appPassword` properties on the WLST command `createSESConnection`.

   To do this with WLST, use the `createFederationTrustedEntity` command (Example 18–1).

Example 18–1 createFederationTrustedEntity Command

```
createFederationTrustedEntity(
appName='webcenter',
sesUrl='http://mySEShost.com:7777/search/api/admin/AdminService',
sesPassword='mySESAdminPassword', entityName='myTrustedEntityUser',
entityPassword='myTrustedEntityUserPassword', desc='Trusted entity for WebCenter
Portal', sesSchema='eqsys')
```

   For command syntax and examples, see the "createFederationTrustedEntity" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

   To do this in Oracle SES, follow these steps.

   a. In the Oracle SES administration tool, navigate to the Global Settings - Federation Trusted Entities page.

   b. Enter a name for a trusted entity. This is the name that WebCenter Portal uses to authenticate itself to Oracle SES at search time (before it propagates the end user identity to Oracle SES).

To allow the entity to be authenticated through the active identity plug-in:

- Make sure that the entity name is the name of a user that exists in the identity management system.

- Leave the password field blank.

- Select the **Use Identity Plug-in for authentication** checkbox.

- Enter the authentication attribute value corresponding to the Authentication Attribute in your active identity plug-in.

To allow the entity to be authenticated through Oracle SES:

- Enter any user name (for example, `wcsearch`) and password (for example, `myPassword1`).

- Do *not* select the **Use Identity Plug-in for authentication** checkbox.

For more information, see the online help for the Federation Trusted Entities page in Oracle SES.

Note: For reference, the following sample user names are used in this chapter:

- `wcsearch`: User of the Oracle SES Federation Trusted Entity
- `mycrawladmin`: Crawl admin user in WebCenter Portal and in the identity management system to crawl certain objects, such as lists, page metadata, portals, and profiles
- `sescrawer` (or admin user): Crawl admin user in Oracle WebCenter Content Server with `sescrawlerrole` (or `admin`) role

---

18.3.3 Oracle SES - Security

Most enterprise deployments require that their HTTP connections be secure. SSL (secure socket layer) is an encryption protocol for securely transmitting private content on the internet. Oracle strongly recommends that you use an SSL-protected channel to transmit password and other secure data over networks.

For instructions, see Section 35.10, "Securing the Connection to Oracle SES with SSL."

18.4 Setting Up Oracle SES Connections

This section includes the following topics:

18.4.1 Testing the Connection to Oracle SES

Confirm the connection to Oracle SES by entering in a browser the URL for Oracle SES Web Services operations:

`http://host:port/search/query/`

If the URL address *does not* render in the browser, then either the host or port for the Oracle SES server is incorrect, or Oracle SES has not been started.

> **Note:** With Oracle SES 11.2.2.2, you must run the `setSESVersion` WLST command to enable the Tools and Services - Search administration page. For more information, see Section 18.5.5, "Configuring Oracle SES Version Using WLST."

18.4.2 Registering Oracle Secure Enterprise Search Servers

You can register multiple Oracle SES connections with a WebCenter Portal application (or your Portal Framework application) but only one of them is active at a time.

You can register Oracle SES connections using either Fusion Middleware Control or WLST. This section includes the following topics:

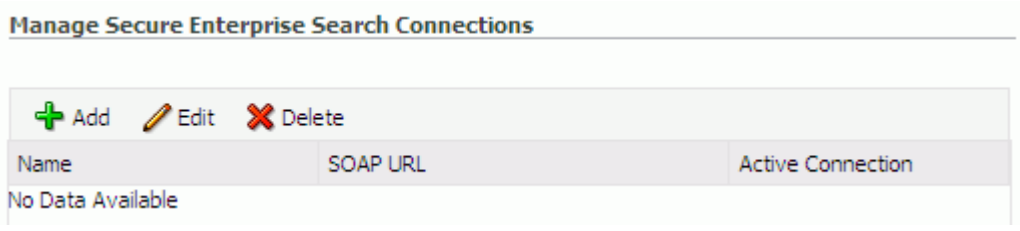
- Section 18.4.2.1, "Registering Oracle SES Connections Using Fusion Middleware Control"
- Section 18.4.2.2, "Registering Oracle SES Connections Using WLST"

18.4.2.1 Registering Oracle SES Connections Using Fusion Middleware Control

To register an Oracle SES instance:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal or your Portal Framework application. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For the WebCenter Portal application - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.
3. On the WebCenter Portal Service Configuration page, select **Search**.
4. To connect to a new Oracle SES instance, click **Add** (Figure 18–3).

Figure 18–3 Configuring Oracle Secure Search



5. On the Add Secure Enterprise Search Connection page, enter a unique name for this connection, and indicate whether this connection is the active (or default) connection for the application (Figure 18–4).

Figure 18–4 Add Secure Enterprise Search Connection

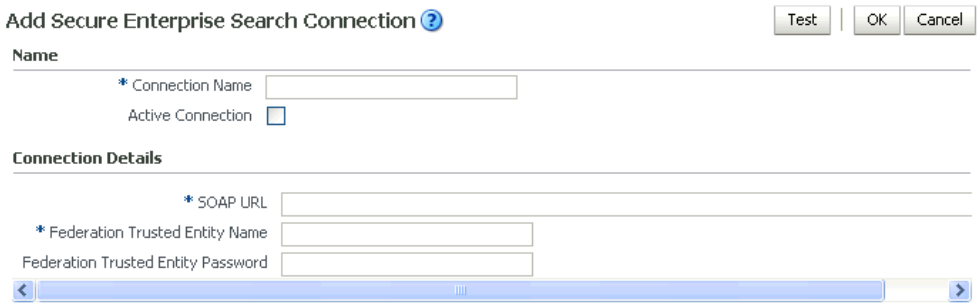


Table 18–3 describes the connection parameters.

Table 18–3 Oracle SES Connection - Name

Field	Description
Connection Name	Enter a unique name for the connection. The name must be unique (across all connection types) within the WebCenter Portal application (or your Portal Framework application).
Active Connection	Select to use the Oracle SES instance defined on this connection as your search platform for WebCenter Portal or your Portal Framework application. While you can register multiple Oracle SES connections for an application, only one connection is used—the default (or active) connection.

6. Enter connection details for the Oracle SES instance (Table 18–4).

Table 18–4 Oracle Secure Enterprise Search - Connection Details

Field	Description

SOAP URL	Enter the Web Services URL that Oracle SES exposes to enable search requests.
	Use the format:
	`http://host:port/search/query/OracleSearch`
	For example:
	`http://myHost:7777/search/query/OracleSearch`

Federation Trusted Entity Name	Enter the user name of the Oracle SES Federation Trusted Entity created in Section 18.3.2, "Oracle SES - Configuration." **Tip:** This user is configured in the Oracle SES administration tool, on the Global Settings - Federation Trusted Entities page. The WebCenter Portal application (or your Portal Framework application) must authenticate itself as a trusted application to Oracle SES to perform searches on behalf of WebCenter Portal users. Examples in this chapter use `wcsearch` for this value.
Federation Trusted Entity Password	Enter the password for the Federation Trusted Entity.

7. Click **OK** to save this connection.

> **Note:** To start using the new (active) connection you must restart the managed server on which the application is deployed (by default, WC_Spaces). For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

18.4.2.2 Registering Oracle SES Connections Using WLST

1. Use the WLST command `createSESConnection` to create a connection to Oracle SES. For example:

```
createSESConnection(appName='webcenter',
                    name='mySesConnection',
                    url='http://myhost.com:7777/search/query/OracleSearch',
                    appUser='wcsearch',
                    appPassword='myPassword1',
                    default=true)
```

   where `appUser` is the user name of the Oracle SES Federation Trusted Entity created in Section 18.3.2, "Oracle SES - Configuration."

For command syntax and examples, see the "createSESConnection" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

To configure search to actively use a new Oracle SES connection, set `default=1` (true). For more information, see Section 18.4.3.2, "Choosing the Active Oracle SES Connection Using WLST."

> **See Also:** Section 18.4.4.2, "Modifying Oracle SES Connection Details Using WLST"

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

> **Note:** To start using the new (active) connection or settings, you must restart the managed server on which the application is deployed (by default, WC_Spaces). See the "Starting and Stopping Managed Servers Using WLST" section in *Oracle Fusion Middleware Administrator's Guide*.

18.4.3 Choosing the Active Oracle SES Connection

You can register multiple Oracle SES connections with a WebCenter Portal application (or your Portal Framework application), but only one connection is active at a time.

> **Note:** These steps in this section are not necessary if you selected the active connection in Section 18.4.2, "Registering Oracle Secure Enterprise Search Servers."

This section includes the following topics:

- Section 18.4.3.1, "Choosing the Active Oracle SES Connection Using Fusion Middleware Control"
- Section 18.4.3.2, "Choosing the Active Oracle SES Connection Using WLST"

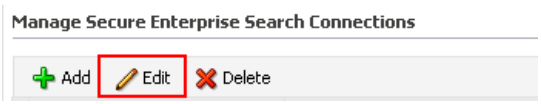
18.4.3.1 Choosing the Active Oracle SES Connection Using Fusion Middleware Control

To change the active connection:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal or your Portal Framework application. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For the WebCenter Portal application - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.
3. On the WebCenter Portal Services Configuration page, select **Search**.

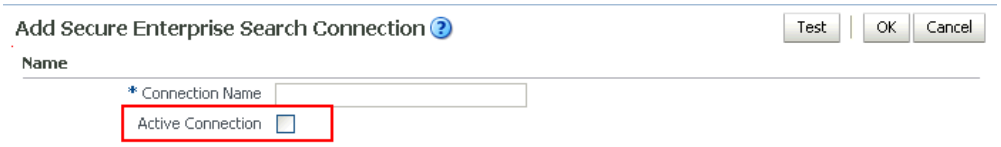
   The Manage Secure Enterprise Search Connections table indicates the current active connection (if any).
4. Select the connection you want to make the active (or default) connection, and then click **Edit** (Figure 18–5).

Figure 18–5 Edit Icon



5. Select the **Active Connection** checkbox (Figure 18–6).

Figure 18–6 Active Connection Checkbox



6. Click **OK** to update the connection.

> **Note:** To start using the new (active) connection you must restart the managed server on which the application is deployed (by default, WC_Spaces). For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

18.4.3.2 Choosing the Active Oracle SES Connection Using WLST

Use the WLST command `setSESConnection` with `default=true` to activate an existing Oracle SES connection. For example:

```
setSESConnection(appName='app1',
                 name='SESConn1',
```

```
url='http://myhost.com:7777/search/query/OracleSearch',
appUser='wpadmin',
appPassword='password',
default=1)
```

For full command syntax and examples, see the "setSESConnection" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

To subsequently disable an Oracle SES connection, run the same WLST command with `default=false`. Connection details are retained but the connection is no longer named as an active connection.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Note: To start using the active connection you must restart the managed server on which the application is deployed (by default, WC_Spaces). For more information, see the "Starting and Stopping Managed Servers Using WLST" section in *Oracle Fusion Middleware Administrator's Guide*.

18.4.4 Modifying Oracle SES Connection Details

You can modify Oracle SES connection details at any time.

To start using the updated (active) connection you must restart the managed server on which the WebCenter Portal application (or your Portal Framework application) is deployed.

Note: The steps in this section are required only to modify the details configured in Section 18.4.2, "Registering Oracle Secure Enterprise Search Servers."

This section includes the following topics:

- Section 18.4.4.1, "Modifying Oracle SES Connection Details Using Fusion Middleware Control"
- Section 18.4.4.2, "Modifying Oracle SES Connection Details Using WLST"

18.4.4.1 Modifying Oracle SES Connection Details Using Fusion Middleware Control

To update connection details for an Oracle SES instance:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal or your Portal Framework application. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For the WebCenter Portal application - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.

- For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.

3. On the WebCenter Portal Service Configuration page, select **Search**.
4. Select the connection name, and click **Edit**.
5. Edit connection details, as required. For detailed parameter information, see Table 18–4.
6. Click **OK** to save your changes.

> **Note:** To start using the updated (active) connection you must restart the managed server on which the application is deployed (by default, WC_Spaces). For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

18.4.4.2 Modifying Oracle SES Connection Details Using WLST

Use the WLST command `setSESConnection` to alter an existing Oracle SES connection. For command syntax and examples, see the "setSESConnection" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

> **Note:** To start using the updated (active) connection you must restart the managed server on which the application is deployed (by default, WC_Spaces). For more information, see the "Starting and Stopping Managed Servers Using WLST" section in *Oracle Fusion Middleware Administrator's Guide*.

18.4.5 Deleting Oracle SES Connections

You can delete Oracle SES connections at any time but take care when deleting the active connection. If you delete the active connection, users are not able to search content on external repositories.

This section includes the following topics:

- Section 18.4.5.1, "Deleting Oracle SES Connections Using Fusion Middleware Control"
- Section 18.4.5.2, "Deleting Oracle SES Connections Using WLST"

18.4.5.1 Deleting Oracle SES Connections Using Fusion Middleware Control

To delete an Oracle SES server connection:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal or your Portal Framework application. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For the WebCenter Portal application - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.

- For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.

3. From the Service Connection drop-down, select **Search**.
4. Select the connection name, and click **Delete**.

Figure 18–7 Delete Connection Icon

> **Note:** To effect this change you must restart the managed server on which the application is deployed (by default, WC_Spaces). For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

18.4.5.2 Deleting Oracle SES Connections Using WLST

Use the WLST command `deleteConnection` to remove an Oracle SES connection. For command syntax and examples, see the "deleteConnection" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

> **Note:** To effect this change you must restart the managed server on which the application is deployed (by default, WC_Spaces). For more information, see the "Starting and Stopping Managed Servers Using WLST" section in *Oracle Fusion Middleware Administrator's Guide*.

18.5 Configuring Oracle SES to Search WebCenter Portal Applications

This section describes the steps necessary to set up Oracle SES for WebCenter Portal applications. It includes the following topics:

- Section 18.5.1, "Setting Up WebCenter Portal for Oracle SES"
- Section 18.5.2, "Setting Up Oracle WebCenter Content Server for Oracle SES"
- Section 18.5.3, "Setting Up Oracle WebCenter Portal Discussion Server for Oracle SES"
- Section 18.5.4, "Setting Up Oracle SES to Search WebCenter Portal"
- Section 18.5.5, "Configuring Oracle SES Version Using WLST"
- Section 18.5.6, "Configuring Search Crawlers Using WLST"
- Section 18.5.7, "Tips for Crawling Page Metadata"

> **Note:** For an overview of the tasks that must be performed to enable Oracle SES as the search engine, see Section 18.2, "Configuration Roadmaps for Oracle SES in WebCenter Portal." There may be various acceptable ways and orders to perform the required tasks.

See Also: Section 18.1, "About Search with Oracle SES"

18.5.1 Setting Up WebCenter Portal for Oracle SES

This section describes how to configure WebCenter Portal to work with Oracle SES.

1. Create and configure the connection between WebCenter Portal and Oracle SES, and optionally specifying a source group.

   See Also: Section 18.4.2.2, "Registering Oracle SES Connections Using WLST" and Section 18.4.4.2, "Modifying Oracle SES Connection Details Using WLST"

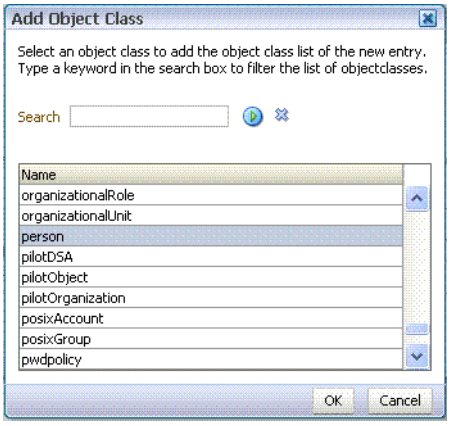
2. To use Oracle SES to search portals, lists, or page metadata, you must first create a *crawl admin user* in WebCenter Portal and in your back-end identity management server (for example, `mycrawladmin`). You must only create a crawl admin user once.

   Note: See your identity management system documentation for information on creating users.

   The following example uses Oracle Directory Services Manager to create the `mycrawladmin` user.

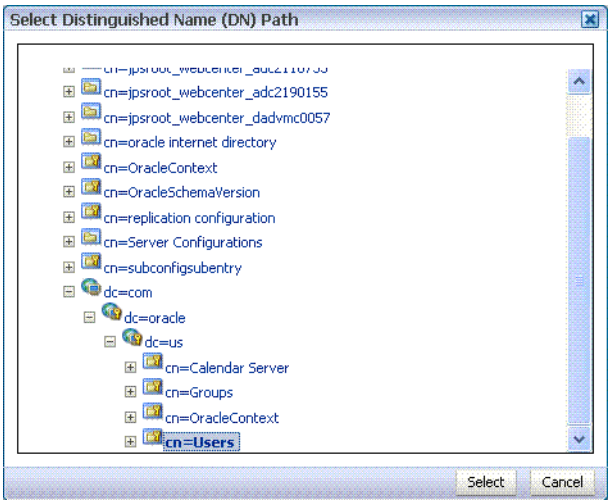
   a. On the Data Browser tab, navigate to the target `cn` and click **Create**. This example navigates to "dc=com,dc=oracle,dc=us,cn=Users". In the Add Object Class dialog, select the appropriate object class, and click **OK**. (Figure 18–8).

Figure 18–8 Oracle Directory Services Manager - Add Object Class



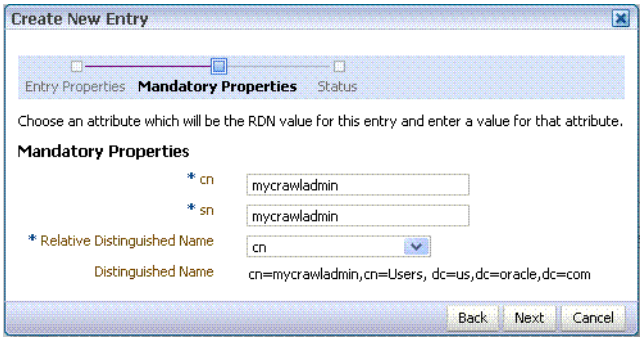
   b. Find the distinguished name (DN) path, and click **Select** (Figure 18–9). This example selects "dc=com,dc=oracle,dc=us,cn=Users".

Figure 18–9 Oracle Directory Services Manager - Select DN Path



c. In the Create New Entry dialog, enter properties, and click **Next** (Figure 18–10).

Figure 18–10 Oracle Directory Services Manager - Create New Entry



d. When you see that the new entry was created successfully, click **Finish**. (Figure 18–11)

Figure 18–11 Oracle Directory Services Manager - Status

3. Grant the crawl application role to the crawl admin user created in Section 18.3.2, "Oracle SES - Configuration." For example:

```
grantAppRole(appStripe="webcenter",
             appRoleName="webcenter#-#defaultcrawl",
             principalClass="weblogic.security.principal.WLSUserImpl",
             principalName="mycrawladmin");
```

For command syntax and examples, see the "grantAppRole" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

> **Note:** To effect WLST changes, you must restart the managed server on which the application is deployed (by default, WC_Spaces). For more information, see the "Starting and Stopping Managed Servers Using WLST" section in *Oracle Fusion Middleware Administrator's Guide*.

4. Enable the Oracle SES crawlers in WebCenter Portal.

   With the same WLST command, you can set crawler properties (that is, enable/disable the crawlers) and specify an interval between full crawls for the spaces crawler. By default, full crawls for the spaces crawler occur every seven days, but you can specify a different frequency. (Incremental crawls are initiated by the schedule set in Oracle SES.)

   For example:

```
setSpacesCrawlProperties(appName='webcenter',
                         fullCrawlIntervalInHours=168,
                         spacesCrawlEnabled = true,
                         documentCrawlEnabled=true,
                         discussionsCrawlEnabled=true)
```

   > **Notes:** The `spacesCrawlEnabled`, `documentCrawlEnabled` and `discussionsCrawlEnabled` parameters all must be set to `true` to enable Oracle SES.
   >
   > A clustered instance additionally requires the `server` parameter; for example, `server="WC_Spaces1"`.

   The following example specifies that WebCenter Portal runs a full crawl through the spaces crawler every 8 days.

```
setSpacesCrawlProperties(appName='webcenter',
                        fullCrawlIntervalInHours=192)
```

   You also can use WLST to return the current crawl settings for WebCenter Portal, such as the number of hours between full crawls (spaces crawler). For example, the following command returns the current crawl settings.

```
getSpacesCrawlProperties(appName='webcenter')

WebCenter Crawl Properties:
-----------------
fullCrawlIntervalInHours: 124
spacesCrawlEnabled:       true
documentCrawlEnabled:     true
discussionsCrawlEnabled:  true
```

5. Use Fusion Middleware Control or the `listDocumentsSpacesProperties` command to determine the unique name that the back-end Content Server is using to identify this WebCenter Portal application and the connection name for the primary Content Server that WebCenter Portal is using to store documents.

   For example:

```
listDocumentsSpacesProperties(appName='webcenter')
```

The response should looks something like the following:

```
The Documents Spaces container is "/WebCenter1109"
The Documents repository administrator is "sysadmin"
The Documents application name is "WC1109"
The Documents primary connection is "stxxl18-ucm11g"
```

Note: Record the application name and the primary connection returned. These values are required later (in Section 18.5.4.2, "Setting Up Oracle SES to Search Documents") to set up Oracle SES to crawl documents.

Note: To effect WLST changes, you must restart the managed server on which the application is deployed (by default, WC_Spaces). For more information, see the "Starting and Stopping Managed Servers Using WLST" section in *Oracle Fusion Middleware Administrator's Guide*.

18.5.1.1 Configuring Search Parameters Using WLST

Use the WLST command `setSearchConfig` to modify search parameters.

Example 18–2 shows how to specify a data group (also known as source group) under which you search Oracle SES.

Example 18–2 Set a Source Group

```
setSearchSESConfig(appName='webcenter',
                   dataGroup='mySources')
```

where `dataGroup` is the source group you create in Section 18.5.4.6, "Additional Oracle SES Configuration."

Example 18–3 shows how to increase the number of search results displayed. Five is the default setting for the number of search results displayed in Oracle SES results, but result sets generally are larger than five.

Example 18–3 Increase Number of Search Results Displayed

```
setSearchConfig(appName='webcenter',
                numResultsMain=10)
```

Example 18–4 shows how to configure the maximum time that a service is allowed to execute a search (in ms). When a service times out largely depends on the system load. If you consistently get time out errors, adjust this parameter.

Example 18–4 Configure Maximum Time WebCenter Portal Waits for Search Results

```
setSearchConfig(appName='webcenter',
                executionTimeout=10000)
```

For command syntax and examples, see the "setSearchConfig" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

18.5.1.2 Configuring Search Parameters and Crawlers Using Fusion Middleware Control

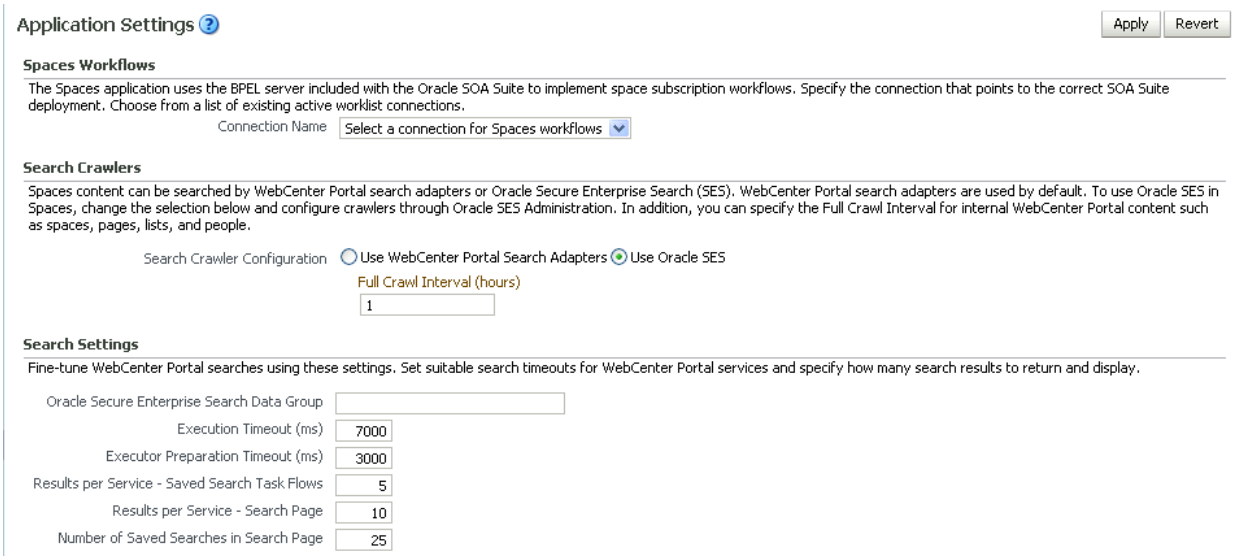
You can enable or disable Oracle SES and configure search settings using Fusion Middleware Control.

1. Log in to Fusion Middleware Control and navigate to the home page for your WebCenter Portal application. For more information, see Section 6.2, "Navigating to the Home Page for WebCenter Portal."
2. From the **WebCenter Portal** menu, select **Settings** > **Application Configuration**.
3. In the **Search Crawlers** section, select to use Oracle SES.

   Optionally, you can specify how often WebCenter Portal content is crawled. By default, full crawls for the spaces crawler occur every seven days, but you can specify a different frequency. (Incremental crawls, for all three crawlers, are initiated by the schedule set in Oracle SES.)

   Click **Apply** (Figure 18–12).

Figure 18–12 Application Settings for WebCenter Portal Search



4. On the same page, configure **Search Settings** parameters as required, and then click **Apply**.
   - **Oracle Secure Enterprise Search Data Group**: Specify the Oracle SES source group in which to search. If no value is provided, then everything in the Oracle SES instance is searched.
   - **Execution Timeout**: Enter the maximum time that a service is allowed to execute a search (in ms).
   - **Executor Preparation Timeout**: Enter the maximum time that a service is allowed to initialize a search (in ms).
   - **Results per Service - Saved Search Task Flows**: Enter the number of search results displayed, per service, in a Saved Search task flow.
   - **Results per Service - Search Page**: Enter the number of search results displayed, per service, for searches submitted from the main search page. Users can click Show All if they want to see all the results.

- **Number of Saved Searches in Search Page**: Enter the number of saved searches displayed in the Saved Search list (on the main search page).

You do *not* need to restart the managed server on which the WebCenter Portal application is deployed.

18.5.2 Setting Up Oracle WebCenter Content Server for Oracle SES

This section describes how to configure Oracle WebCenter Content Server to be crawlable by Oracle SES (in particular, the Content Server that WebCenter Portal uses for storing documents).

The following steps must be done from within the Content Server.

> **See Also:** Content Server online help for information on administering roles and users in Content Server

1. Create a crawl user.

   If you want users with the `admin` role to crawl, then use an admin user account as the crawl user.

   If you want non-admin users to crawl, then follow these steps:

   a. Create the role `sescrawlerrole`.

   b. Create the user `sescrawler`, and assign it the `sescrawlerrole` role. This user creates the Content Server source in Oracle SES.

   c. Add `sceCrawlerRole=sescrawlerrole` to `config.cfg` (located in *`MW_HOME`*`/user_projects/domains/`*`yourdomain`*`/ucm/cs/config`).

      Alternatively, you can append the `sceCrawlerRole=sescrawlerrole` line in the WebCenter Content Server user interface (Administration - General Configuration - Additional Configuration Variables).

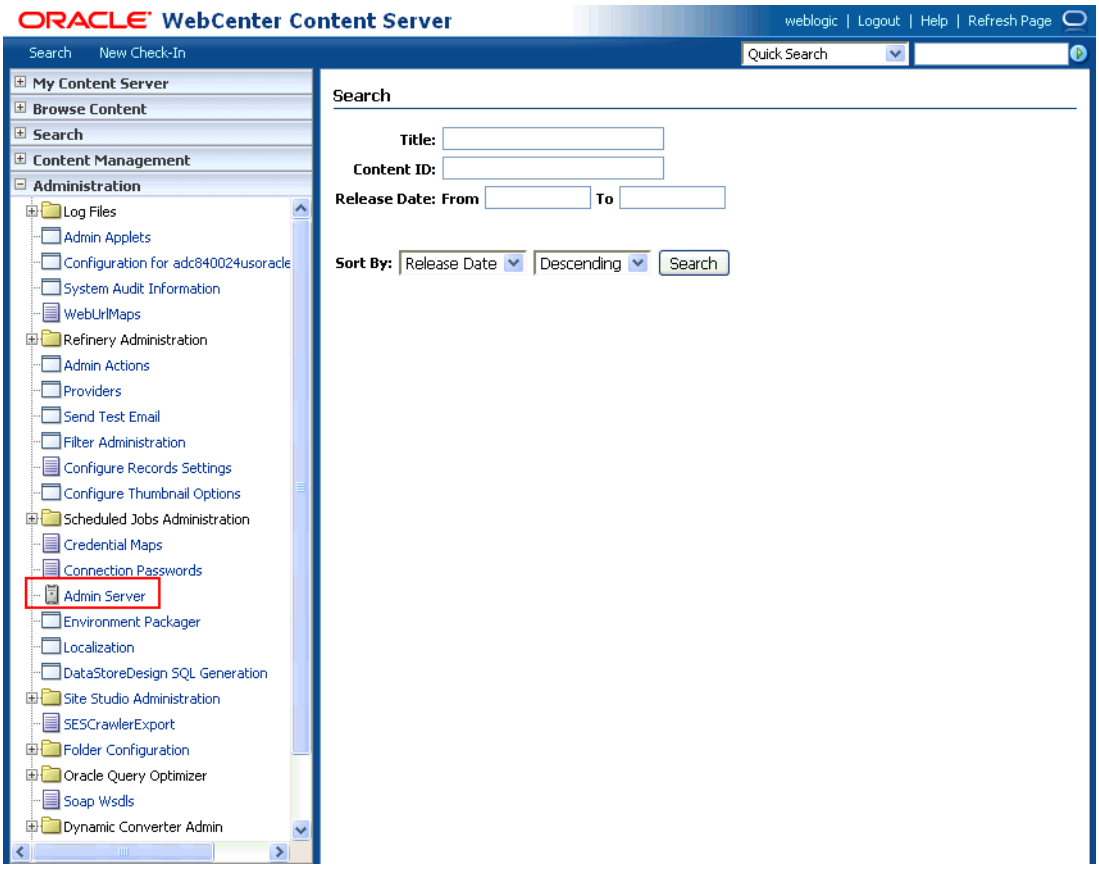
2. Restart the Content Server.

3. In the Content Server console, install the SESCrawlerExport component on the content server, if not done:

   a. Log on to the Content Server as a system administrator. For example: `http://`*`host:port`*`/cs`.

   b. From the Administration dropdown menu, select **Admin Server** (Figure 18–13).

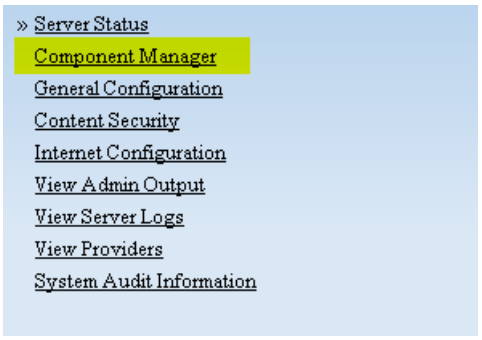
Figure 18–13 Content Server Administration



c. Click the button with the instance name.

d. Click **Component Manager** from the menu list on the left pane (Figure 18–14).

Figure 18–14 Content Server Component Manager



e. Select **SESCrawlerExport** under Integration and click **Update**.

f. Enter configuration parameters. (You can change configuration parameters after installation.)

Disable security on authentication and authorization APIs provided by the SESCrawlerExport; that is, set **Disable Secure APIs** to `false`. This lets security provided by the SESCrawlerExport be done internally instead of by the content server.

Additionally, in clustered environments only, the **feedLoc** parameter must specify a location on the shared disk accessed by the nodes of the Content Server, and they each must reference it the same way; for example, `sharedDrive/dir1/dir2`. Note that this is not the default location (relative path) provided.

g. Restart the Content Server.

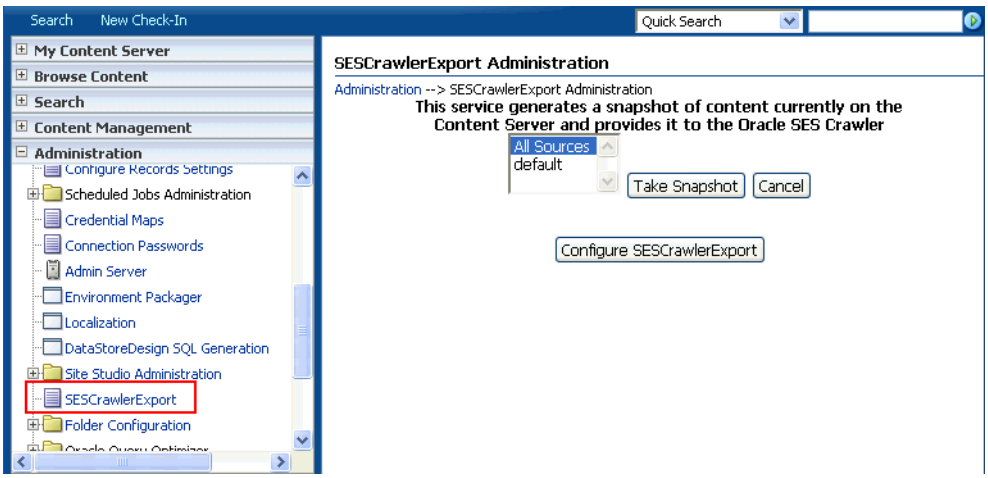
4. Take a snapshot of the Content Server repository.

   a. Log on to the Content Server as a system administrator. For example: `http://host:port/cs`.

   b. From the Administration dropdown menu, select **SESCrawlerExport**.

   c. Select **All sources**, and click **Take Snapshot** (Figure 18–15).

Figure 18–15 Content Server Snapshot



It is important to take a snapshot before the first crawl or any subsequent full crawl of the source.

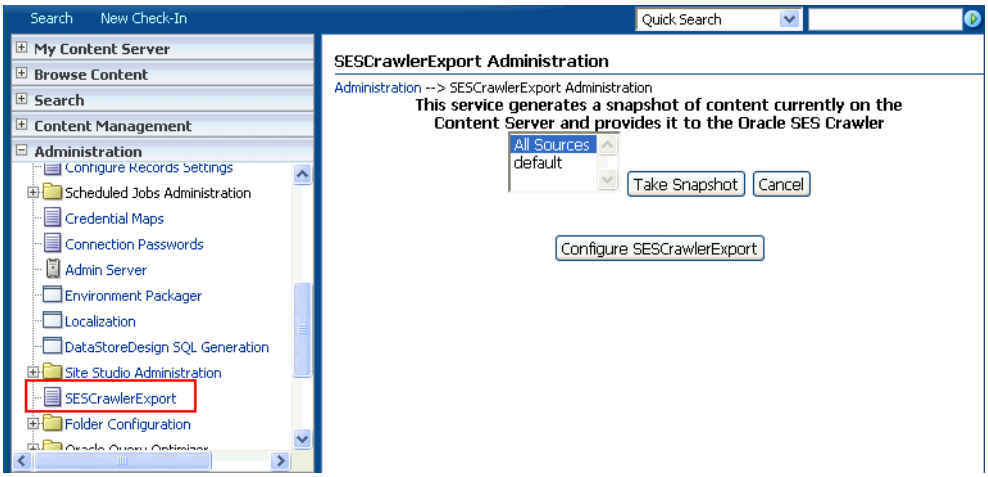
The snapshot generates `configFile.xml` at the location specified during component installation, and feeds are created at the subdirectory with the source name under **feedLoc**.

5. If the Content Server is configured for web rendition, then items in the Content Server are rendered in PDF format. The content item's native MIME type rendition is overwritten. For example, the MIME type of a Microsoft Office Word document is 'application/msword', but when the Content Server uses web rendition the MIME type becomes 'application/pdf'. A search query with the `Mimetype` parameter set to 'application/msword' does not return Word documents.

   If your Content Server is configured to use web rendition, then you must configure the Content Server metatdata list to include the `dFormat` value, so that required MIME types are exported to Oracle SES. This is necessary to be able to narrow searches by MIME type.

   a. Back on the SESCrawlerExport Administration page, click **Configure SESCrawlerExport** (Figure 18–31).

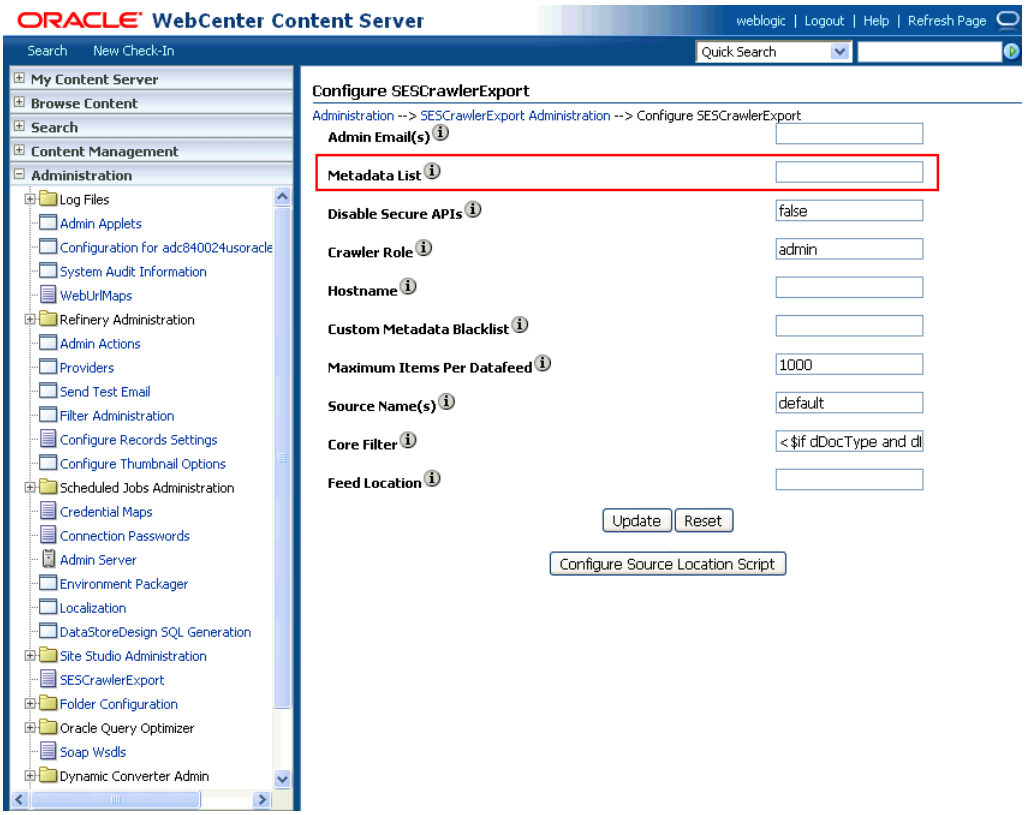
Figure 18–16 Content Server Snapshot



b. By default, the **Metadata List** field is blank (Figure 18–17). Optionally, add to this field any custom metadata values you require (beginning with `x`). For example, the following entry for **Metadata List** includes custom attributes:

```
xCollectionID,xWCTags, xRegionDefinition
```

Figure 18–17 Content Server Metadata List



c. If you are using a version of Content Server *earlier than* 11.1.1.8, then you must add the `dFormat` value to this **Metadata List** field. When the blank default value is changed, the default values are removed, and they must be added

back. (Content Server 11.1.1.8 includes most values required for search, including `dFormat`.)

Therefore, if you are using a version of Content Server earlier than 11.1.1.8, you must manually enter the default value for **Metadata List** as follows (plus any custom metadata fields beginning with 'x'):

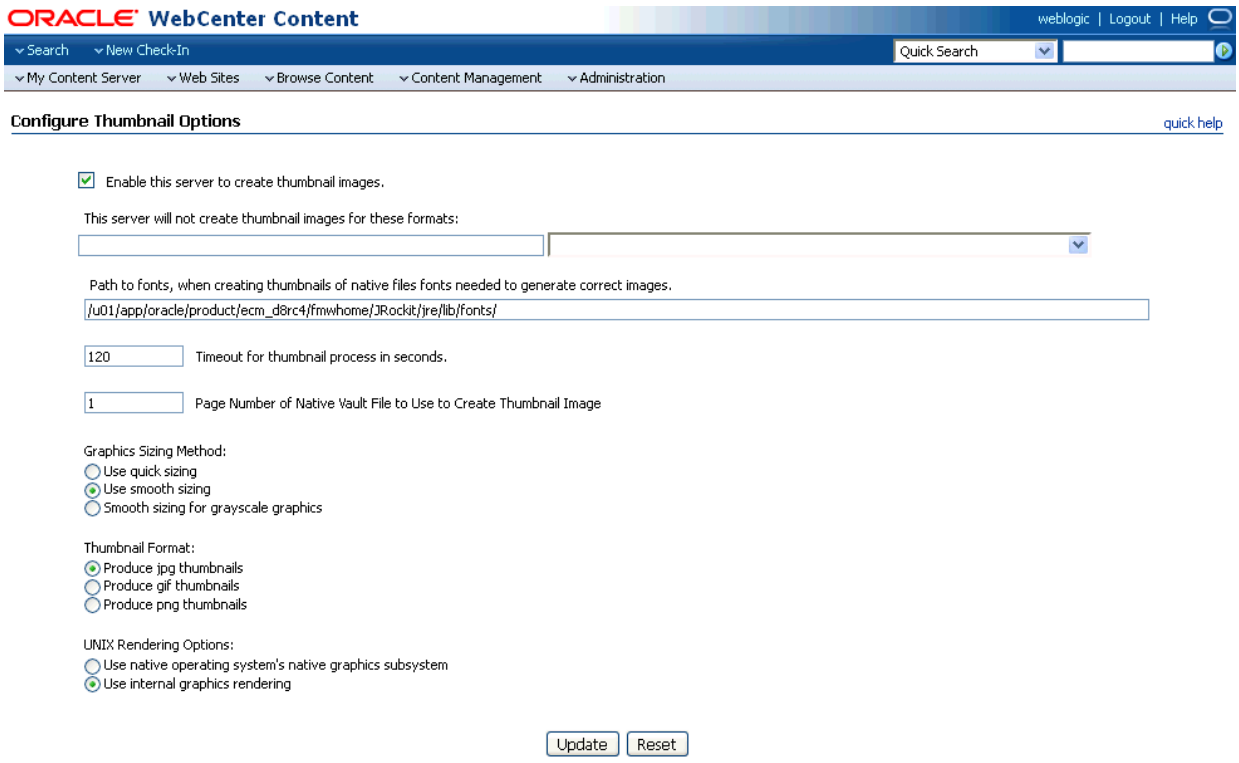
```
dFormat,dID,dDocName,dRevLabel,dDocType,dDocAccount,dSecurityGroup,
dOriginalName,dReleaseDate,dOutDate,dDocCreator,dDocLastModifier,dDocCreate
dDate,dDocFunction
```

6. Configure Thumbnail Options for faceted search.

Note: Oracle SES 11.2.2.2 supports document thumbnails, while earlier releases of Oracle SES do not.

On the **Administration** tab, select **Configure Thumbnail Options** to enable document thumbnails in search results. Leave the default settings as is, and click **Update** (Figure 18–18).

Figure 18–18 Configure Thumbnail Options



See Also: The `Deployment Guide.pdf` included with the product for detailed information on Content Server configuration

18.5.3 Setting Up Oracle WebCenter Portal Discussion Server for Oracle SES

This section describes how to configure Oracle WebCenter Portal's Discussion Server to be crawlable by Oracle SES (in particular, the discussions server that WebCenter Portal uses for storing discussions and announcements).

Note: These steps are not required if you have a new installation of WebCenter Portal (with an Oracle database) and Oracle WebCenter Portal's Discussion Server. It is only required if you are using upgraded (patched) instances.

You can find database schema details for the corresponding data sources from your Oracle WebLogic Server console.

---

1. Run the Repository Creation Utility (RCU) to confirm that the discussions crawler WebCenter Portal component has been installed on the system.

   - Oracle and Microsoft SQL Server databases:

     Verify that the Oracle WebCenter Portal's Discussion Server back end has been configured properly by noting that the *MyPrefix*_DISCUSSIONS user is installed in RCU.

     Then verify that the discussions crawler has been configured properly by noting that the *MyPrefix*_DISCUSSIONS_CRAWLER user is installed in RCU.

   - IBM DB2 databases:

     Verify that the Oracle WebCenter Portal's Discussion Server back end has been configured properly by noting that the *MyPrefix*_DS user is installed in RCU.

     Then verify that the discussions crawler has been configured properly by noting that the *MyPrefix*_DC user is installed in RCU.

     Note: For IBM DB2 databases, *MyPrefix* is limited to five characters. IBM DB2 uses operating system users for authentication (that is, the database user is actually an operating system user). Because some operating systems have an eight character limit for user names, this requires an eight character user name.

   If the discussions crawler component is not installed, then you must install it using RCU, selecting the same prefix that was used for the Oracle WebCenter Portal's Discussion Server component. Also, during the tablespace specification step in RCU, select *Prefix*_IAS_DISCUSSIONS as the default tablespace. This installs the user for Oracle SES.

   For more information, see Chapter 42, "Deploying Portal Framework Applications."

2. For instances upgraded from WebCenter 11.1.1.1.0 only, run the following tool to upgrade the data in the Oracle WebCenter Portal's Discussion Server database schema, if you have not run the tool yet:

   Note: This step is necessary only if the instance is upgraded from WebCenter 11.1.1.1.0. For instances installed after WebCenter 11.1.1.1.0, this is not required.

```
java -jar \
$MW_HOME/discussionserver/discussionserver-upgradeforses.jar \
<command_line_parameters>
```

where *command_line_parameters* are the following MDS schema details and discussions database schema details:

```
-mds_jdbc_user user_id \
-mds_jdbc_password password \
-mds_jdbc_url url \
-discussions_jdbc_user user_id \
-discussions_jdbc_password password \
-discussions_jdbc_url url
```

where *mds_jdbc_user*, *mds_jdbc_password*, and *mds_jdbc_url* are the values to log in to the MDS schema, and *discussions_jdbc_user*, *discussions_jdbc_password*, and *discussions_jdbc_url* are the values to log in to the discussions database schema.

For example:

```
java -jar
$MW_HOME/as11r1wc/discussionserver/discussionserver-upgradeforses.jar\
-mds_jdbc_user foo \
-mds_jdbc_password myPassword1 \
-mds_jdbc_url jdbc:oracle:thin:@host:port:SID \
-discussions_jdbc_user foo \
-discussions_jdbc_password myPassword1 \
-discussions_jdbc_url jdbc:oracle:thin:@host:port:SID
```

18.5.4 Setting Up Oracle SES to Search WebCenter Portal

The steps in this section must be performed in the Oracle SES administration tool.

The following steps are required:

1. Section 18.5.4.1, "Logging on to the Oracle SES Administration Tool"
2. Section 18.5.4.2, "Setting Up Oracle SES to Search Documents"
3. Section 18.5.4.3, "Setting Up Oracle SES to Search Discussions and Announcements"
4. Section 18.5.4.4, "Setting Up Oracle SES to Search Portals, Lists, People, and Page Metadata"
5. Section 18.5.4.7, "Configuring Oracle SES Facets and Sorting Attributes"
6. Section 18.5.4.6, "Additional Oracle SES Configuration"

> **See Also:** Confirm that you have installed all required patches for Oracle SES. For the latest information on required patches, see the "Back-End Requirements for Search" section in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal* and the Release Notes.
>
> For detailed information about Oracle SES configuration, see the Oracle SES documentation included with the product. (This is listed in the WebCenter Portal product area on the Oracle Fusion Middleware documentation library.)

18.5.4.1 Logging on to the Oracle SES Administration Tool

Open the Oracle SES administration tool:

1. Open a browser and enter the URL provided after the Oracle SES installation. (This has the form `http://host:port/search/admin/index.jsp`.)

2. Log on with the Oracle SES admin user name and the password specified during installation.
   - For **Oracle SES 11.2.2.2**, the default admin user name is `searchsys`; however, a different name may be specified during installation.
   - For **Oracle SES 11.1.***, the admin user name is `eqsys`.

18.5.4.2 Setting Up Oracle SES to Search Documents

To search WebCenter Portal documents using Oracle SES, you must first set up a Document Service Manager (with a Document Service Instance and a Document Service Pipeline), and then create a Content Server source.

1. Configure the Document Service Manager (one time for each Oracle SES instance).

   Note: Document services are plug-ins involved in the processing of a document when it is being crawled. A document service allows WebCenter Portal to add indexable attributes for documents used in a WebCenter Portal application.

   Search attribute names must be unique; two attributes cannot have the same name. For example, if an attribute exists with a String data type, and another attribute is discovered by the crawler with the same name but a different data type, then the crawler ignores the second attribute. Before creating new attributes, make sure to check the list of Oracle SES attribute names and types in the Oracle SES documentation.

   a. On the Global Settings - Document Services page, click **Create**. Select **Create New Manager**, click **Next**, and enter the following parameters:

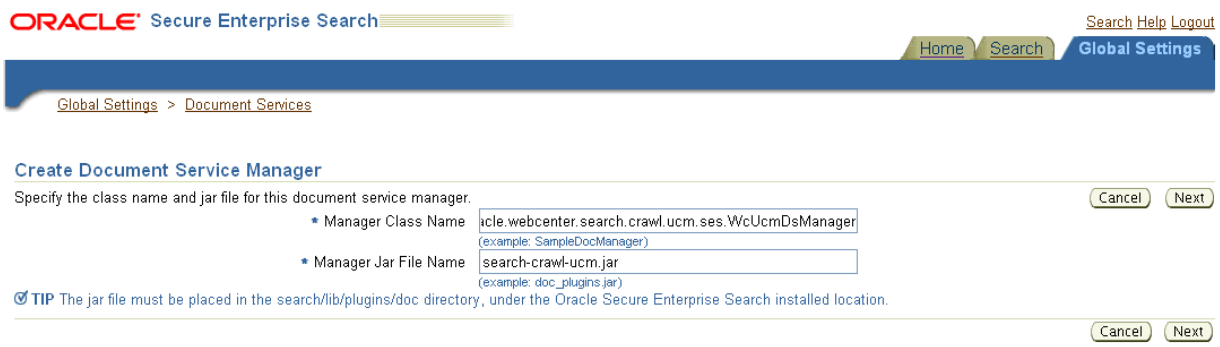
      Manager Class Name: `oracle.webcenter.search.crawl.ucm.ses.WcUcmDsManager`

      Manager Jar File Name: `search-crawl-ucm.jar`

      Note: The `webcenter_doc_pipeline_plugin.zip` file installs `Oracle_Home/search/lib/plugins/doc/search-crawl-ucm.jar`.

      Click **Next**, and then click **Finish** (Figure 18–19).

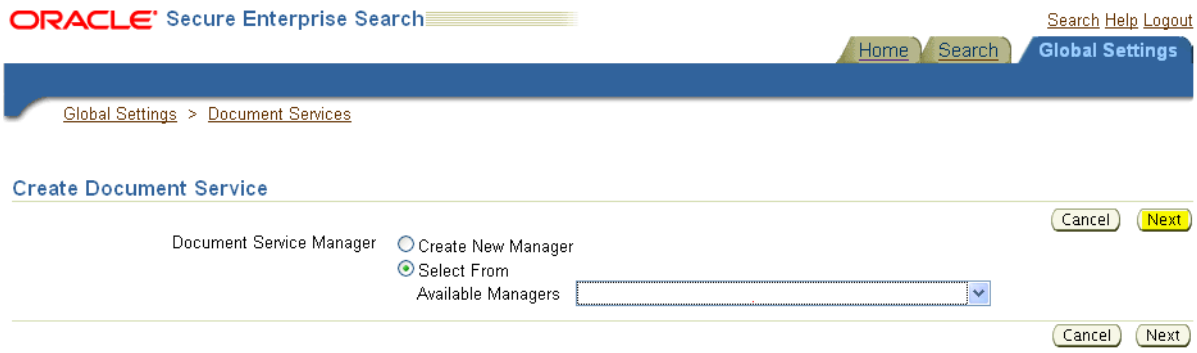
Figure 18–19 Creating a Document Service Manager in Oracle SES



b. Create the Document Service Instance.

Again, on the Global Settings - Document Services page, click **Create.** This time, select **Select From Available Managers** with **Secure Enterprise Search WebCenter UCM Plugin**, and click **Next** (Figure 18–20).

Figure 18–20 Create Document Service



Enter the following parameters:

Instance Name: Enter any name here to be used while creating the document pipeline.

WebCenter Application Name: The unique name being used to identify this WebCenter Portal application in the back-end Content Server.

Connection Name: The name of the primary Content Server connection that WebCenter Portal is using to store documents.

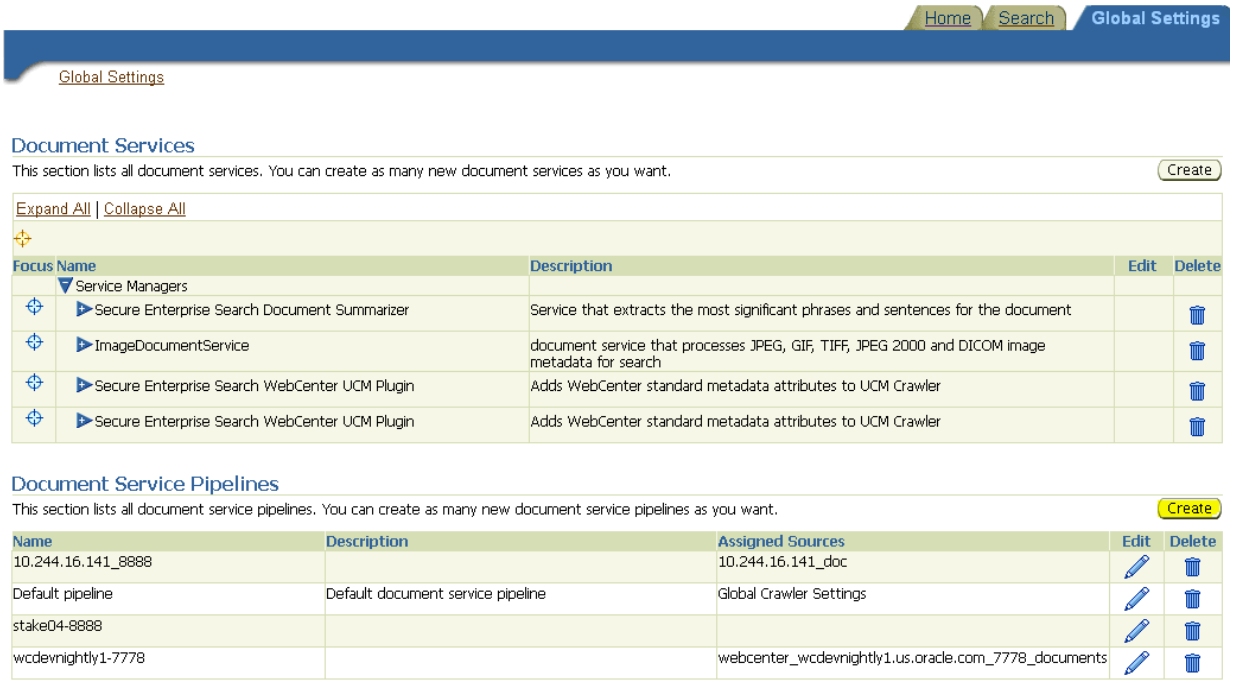
WebCenter URL Prefix: The host and port where the WebCenter Portal application is deployed; for example: `http://myhost:8888`.

Note: Use Fusion Middleware Control or the `listDocumentsSpacesProperties` command to determine the application name and connection name, as described in Section 18.5.1, "Setting Up WebCenter Portal for Oracle SES."

---

c. Create the Document Service Pipeline. This invokes the document service instance. Again, on the Global Settings - Document Services page, under the **Document Services Pipelines** section, click **Create** (Figure 18–21).

Figure 18–21 Creating the Document Service Pipeline



d. On the Create Document Service Pipeline page, enter any custom name for this pipeline. The document service instance you created in the previous step should be listed under **Available Services**. Select that document service instance, and use the arrow button to move it under **Used in pipeline**.

2. Create the Content Server source for documents.

See Also: Section 18.5.6, "Configuring Search Crawlers Using WLST" for an alternative way to create the Content Server source

a. Go to **Home > Sources**.

b. From the Source Type dropdown list, select **Oracle Content Server**, and click **Create** (Figure 18–22).

Figure 18–22 Create Oracle Content Server Source

c. Enter the following parameters:

Source Name: *unique_name*

Configuration URL: *Content_Server_SES_Crawler_Export_endpoint*; for example,

`http://host:port/cs/idcplg?IdcService=SES_CRAWLER_DOWNLOAD_CONFIG&source=default`

Note: The `source=default` parameter denotes the name of the source created in the configuration of the SES Crawler Export. The default one is created automatically and called "default."

Authentication Type:

by SSO, then enter `NATIVE`.

If the Content Server is protected by Oracle SSO, then enter `ORASSO`.

User ID: The user to crawl the Content Server must have the `sceCrawlerRole` role defined. The `sceCrawlerRole` is a configuration parameter in `SESCrawlerExport`. Typically, administrators create a special role, assign it no privileges to view content, then create a user account that has this role.

If you do not set up a specific `sceCrawlerRole`, then admin credentials are required to crawl. The `sysadmin` user ID works by default.

If Authentication Type is `ORASSO`, then enter a user ID (and password) of a user in the identity management server fronted by Oracle SSO. This user must have been granted the same permissions as sysadmin. If it is not possible to grant those permissions, then delete the "remote" user corresponding to this user in the Content Server, and create a "local" version of the user (same name) in the Content Server.

Password: Password for this Content Server user.

Realm:

If Authentication Type is `NATIVE`, then enter `Idc Security /cs/idcplg`, where `/cs/` is the context root you provided when you installing the Content Server.

If Authentication Type is `ORASSO`, then leave this parameter blank.

Scratch Directory: Optional. Specify a directory on the system under which the Oracle SES instance resides.

Oracle SSO Login URL:

If Authentication Type is `ORASSO`, then specify a value for Oracle SSO. For example:
`https://login.oracle.com/mysso/signon.jsp?site2pstoretoken=`

If Authentication Type is `NATIVE`, then leave this field blank.

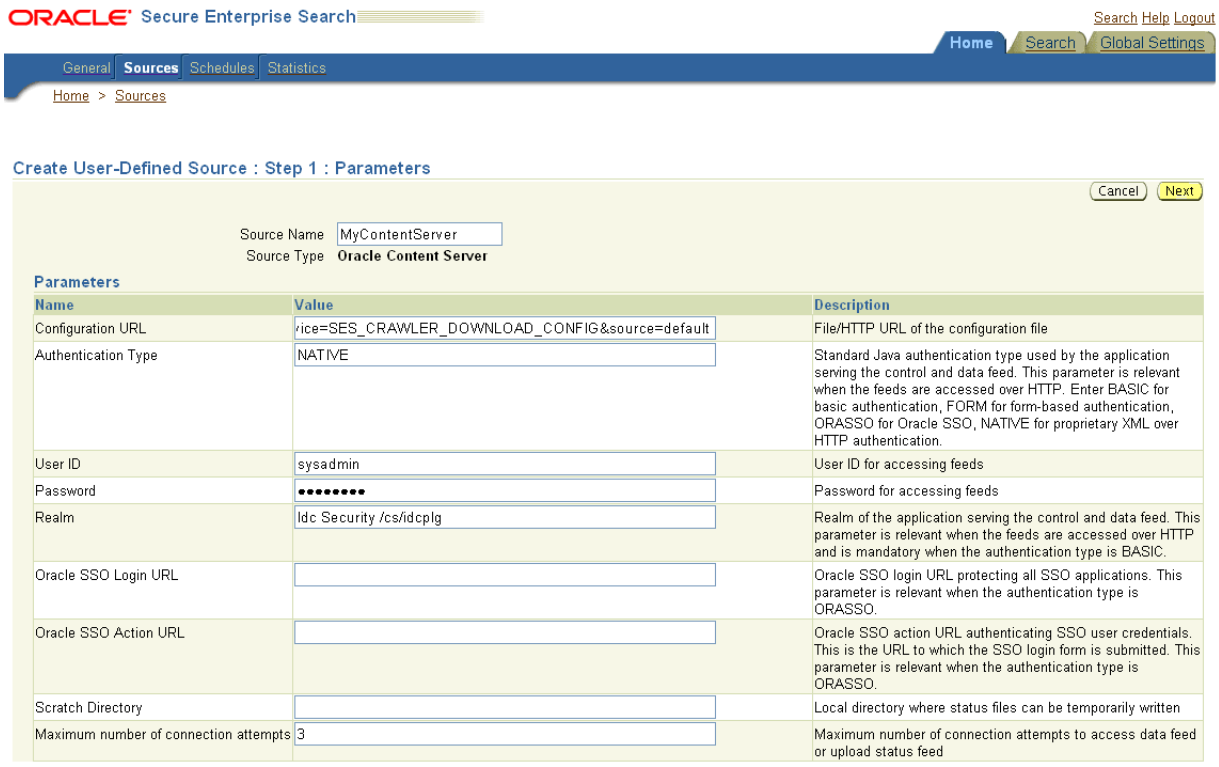
Oracle SSO Action URL:

If Authentication Type is `ORASSO`, then specify a value for Oracle SSO. For example: `https://login.oracle.com/sso/auth`

If Authentication Type is `NATIVE`, then leave this field blank.

Click **Next** (Figure 18–23).

Figure 18–23 Oracle Content Server Source Parameters



d. On the Create User-Defined Source : Step 2 : Authorization page, enter the following parameters in the Authorization Manager section, if not entered by default:

Plug-in Class Name: `oracle.search.plugin.security.auth.stellent.StellentAuthManager`

Jar File Name: `oracleapplications/StellentCrawler.jar`

HTTP endpoint for authorization: for example, `http://host:port/cs/idcplg`

Display URL Prefix: for example, `http://host:port/cs`

Authentication Type: `NATIVE` or `ORASSO`

Administrator User: The user to crawl the Content Server must have the `sceCrawlerRole` role defined. The `sceCrawlerRole` is a configuration parameter in `SESCrawlerExport`. Typically, administrators create a special role, assign it no privileges to view content, then create a user account that has this role.

If you do not set up a specific `sceCrawlerRole`, then admin credentials are required to crawl. The `sysadmin` user ID works by default.

If Authentication Type is `ORASSO`, then enter a user ID (and password) of a user in the identity management server fronted by Oracle SSO. This user must have been granted the same permissions as sysadmin. If it is not possible to grant those permissions, then delete the "remote" user corresponding to this user in the Content Server, and create a "local" version of the user (same name) in the Content Server.

Administrator Password: Password for crawl admin user

Authorization User ID Format: Authentication attribute used in the active identity plug-in. To find this value, go to the Global Settings - Identity Management Setup page in Oracle SES. Enter the value of the Authentication Attribute under the Active Plug-in (for example, `nickname` or `username` or something else). If you are using the Oracle E-Business Suite R12 identity plug-in, then leave the this parameter blank.

Realm:

If Authentication Type is `NATIVE`, then enter `Idc Security /cs/idcplg`, where `/cs/` is the context root you provided when you installing the Content Server.

In Authentication Type is `ORASSO`, then leave this field blank.

e. Click **Create & Customize** (or edit a created source) to see other source parameters. On the **Crawling Parameters** tab, enter the following crawling parameter: `Document Service Pipeline`.

f. Click **Enable** and select the pipeline you created.

18.5.4.3 Setting Up Oracle SES to Search Discussions and Announcements

To search WebCenter Portal discussions and announcements using Oracle SES, you must first set up several Oracle SES Database sources: three for discussions and one for announcements. The three discussions sources are for forums, topics in forums, and replies in forums. These separate sources enable users to see search results for forums without also seeing results for all the messages and replies in it.

For example, the discussions sources could have the following:

- source name `GS_Forums` and View of `FORUMCRAWLER_VW`
- source name of `GS_Topics` and View of `THREADCRAWLER_VW`
- source name of `GS_Replies` and View of `MESSAGECRAWLER_VW`

The announcements source could have the source name `GS_Announcements` and a View of `ANNOUNCEMENTS_VW`.

Note: There are slightly different steps for Oracle, Microsoft SQL Server, and IBM DB2 databases.

---

1. Configure the JDBC driver:

   a. To crawl a Microsoft SQL Server or IBM DB2 database, download the appropriate JDBC driver jar files into the *ORACLE_HOME*`/search/lib/plugins/oracleapplications` directory in Oracle SES.

Note:

- For Microsoft SQL Server: Copy the Microsoft JDBC driver files `sqljdbc.jar` and `sqljdbc4.jar`.
- DB2: Copy the IBM driver files `db2jcc.jar` and `db2jcc_license_cu.jar` (obtainable from the IBM DB2 UDB client).

If the JDBC drivers for JRE 1.5 and JRE 1.6 are different, (for example: `sqljdbc.jar` works for JRE 1.5 and `sqljdbc4.jar` works for JRE 1.6), then perform the following:

- Download both the driver jars into the *ORACLE_HOME*`/search/lib/plugins/oracleapplications` directory in Oracle SES.

- Add an entry for the JRE 1.6 version (`sqljdbc4.jar` for SQLServer) of the driver jar to the `CLASSPATH` element of *ORACLE_HOME*`/search/config/searchctl.conf`.

- Restart the middle tier.

b. Update the `drivers.properties` file with the following information: *`DatabaseName:DriverClassName`*.

c. Add the JRE 1.5 JDBC driver jar file name to the classpath in `META-INF/MANIFEST.MF` of `appsjdbc.jar` and `DBCrawler.jar`.

For example, change:

```
Class-Path:  sqljdbc.jar rsscrawler.jar  ../../pluginmessages.jar
```

to

```
Class-Path:  db2jcc.jar sqljdbc.jar rsscrawler.jar
../../pluginmessages.jar
```

and change:

```
Class-Path: appsjdbc.jar
```

to

```
Class-Path: db2jcc.jar appsjdbc.jar
```

For a `key` attribute that is not named `KEY`, change the JDBC driver information in the `drivers.properties` file to specify the key attribute name: *`database_name: driver_class_name, key_attribute_name`*

For example, for a key attribute named `ID`:

```
oracle : oracle.jdbc.driver.OracleDriver, ID
```

In the crawling query, use *`key_attribute_name`* as the alias for the key value column name. In this example, `ID` is the alias for `KEYVAL`:

```
SELECT keyval id, content, url, lastmodifieddate, lang FROM sales_only
```

Oracle and SQL Server databases: The following default drivers are used if none is specified in `drivers.properties`:

- Oracle: `oracle.jdbc.driver.OracleDriver`
- SQL Server: `com.microsoft.sqlserver.jdbc.SQLServerDriver`

2. Required for IBM DB2 databases only:

a. Make sure that no crawlers are running that use the database crawler source. In the Oracle SES administration tool, check the crawler progress and status on the Home - Schedules page. (Click **Refresh Status**.)

b. Remake the `appsjdbc.jar` file and the `DBCrawler.jar` file. Ensure that the `META-INF/MANIFEST.MF` was updated correctly; otherwise, the crawler fails with the following error in the crawler log file:

```
EQP-80406: Loading JDBC driver failed
```

c. Modify the `ORACLE_HOME/search/lib/plugins/oracleapplications/drivers.properties` file to include the following line:

```
db2: com.ibm.db2.jcc.DB2Driver
```

d. Include the driver jar (`db2jcc.jar`) to the `CLASSPATH` element of `ORACLE_HOME/search/config/searchctl.conf`. For example:

```
#CLASS PATH
CLASSPATH=ORACLE_HOME/search/webapp/config:ORACLE_HOME/search/webapp/
SESAuthenticator.jar:ORACLE_HOME/search/lib/plugins/commons-plugins-
stubs.jar :ORACLE_HOME /search/lib/plugins/oracleapplications/db2jcc.jar
```

e. Edit `JVM_OPTIONS` in the `$ORACLE_HOME/search/config/searchctl.conf` file to add the system property `"-Doracle.home=ORACLE_HOME/search"`. For example:

```
JVM_OPTIONS= -Djava.awt.headless=true
-Dweblogic.RootDirectory=ORACLE_HOME/search/base_domain
-Doracle.home=ORACLE_HOME/search
```

f. Copy the `$ORACLE_HOME/search/lib/plugins/oracleapplications/pluginmessages.jar` file to the `$ORACLE_HOME/search/lib` directory.

g. Create the database source. Make sure to enter the correct authorization query and confirm that the attribute name used in **Grant Security Attributes** matches the one used in the authorization query; otherwise, users do not get any results when searching for documents.

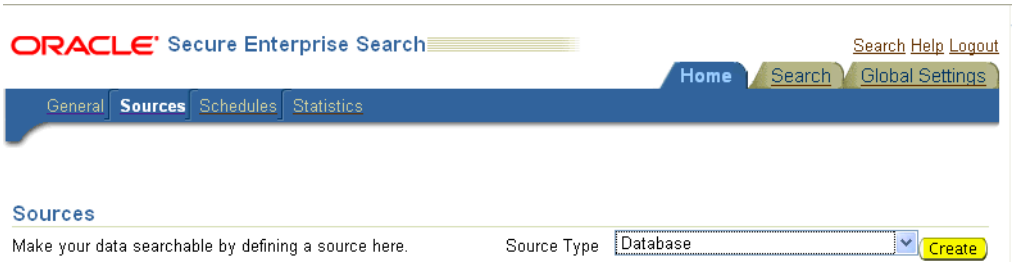
3. Create the Discussions sources or the Announcements source.

See Also: Section 18.5.6, "Configuring Search Crawlers Using WLST" for an alternative way to create these sources

a. In Oracle SES, go to **Home > Sources**.

b. From the Source Type dropdown list, select **Database**, and click **Create** (Figure 18–24).

Figure 18–24 Create Database Source



c. Enter the following parameters:

Source Name: *unique_name*; for example, GS_Forums to crawl discussion forums (or GS_Announcements to crawl announcements)

Database Connection String: Enter one of the following

- Oracle database: Enter one of the following

jdbc:oracle:thin:@*host:port:sid*

jdbc:oracle:thin@*host:port/serviceId*

- IBM DB2 database: Enter jdbc:db2://*host:port/database_name*

- Microsoft SQL Server database: Enter jdbc:sqlserver://*host_or_IP_address:port;database_name*

User ID: Enter one of the following

- Oracle database: The user *MyPrefix*_DISCUSSIONS_CRAWLER created during Oracle WebCenter Portal's Discussions Server installation

- Microsoft SQL Server database: The user *MyPrefix*_DISCUSSIONS_CRAWLER created during Oracle WebCenter Portal's Discussions Server installation

- IBM DB2 database: The user *MyPrefix*_DC created during Oracle WebCenter Portal's Discussions Server installation (where *MyPrefix* is five characters)

Password: Password for this user

Query: Enter one of the following queries:

```
SELECT * FROM FORUMCRAWLER_VW
SELECT * FROM THREADCRAWLER_VW
SELECT * FROM MESSAGECRAWLER_VW
SELECT * FROM ANNOUNCECRAWLER_VW
```

Use FORUMCRAWLER_VW for the source crawling discussion forums.

Use THREADCRAWLER_VW for the source crawling topics in discussion forums.

Use MESSAGECRAWLER_VW for the source crawling replies in discussion forums.

Use ANNOUNCECRAWLER_VW for the source crawling announcements.

URL Prefix: The URL prefix for the WebCenter Portal application, including host, port, and application name. For example, http://*host:port*/webcenter.

Grant Security Attributes: WCSECATTR

Note: Previous releases of Content Server used FORUMID for **Grant Security Attributes**.

---

d. Click **Next**.

e. On the Create User-Defined Source : Step 2 : Authorization page, enter the following parameters (if not prepopulated) in the Authorization Manager section:

Plug-in Class Name:
oracle.search.plugin.security.auth.db.DBAuthManager

Jar File Name: `oracleapplications/DBCrawler.jar`

Authorization Database Connection String: Enter one of the following:

- Oracle database: Enter one of the following:

`jdbc:oracle:thin:@`*`host:port:sid`*

`jdbc:oracle:thin@`*`host:port/serviceId`*

- IBM DB2 database: Enter `jdbc:db2://`*`host:port/database_name`*

- Microsoft SQL Server database: Enter `jdbc:sqlserver://`*`host_or_IP_address:port;database_name`*

User ID: Enter one of the following:

- Oracle database: Enter the user *`MyPrefix`*`_DISCUSSIONS_CRAWLER`

- Microsoft SQL Server database: Enter the user *`MyPrefix`*`_DISCUSSIONS_CRAWLER`

- IBM DB2 database: Enter the user *`MyPrefix`*`_DC` (where *`MyPrefix`* is five characters)

Password: This user password

Single Record Query: `false`

Authorization Query: Enter the following (on one line):

```
SELECT DISTINCT forumID as WCSECATTR
FROM AUTHCRAWLER_FORUM_VW
WHERE username = LOWER(?) UNION SELECT DISTINCT -1 as WCSECATTR
FROM AUTHCRAWLER_FORUM_VW
```

Note: Previous releases of Content Server used the following authorization query:

```
SELECT forumID
FROM AUTHCRAWLER_FORUM_VW
WHERE (username = ? or userID=-1)
UNION SELECT f.forumID
FROM jiveForum f, AUTHCRAWLER_CATEGORY_VW c
WHERE f.categoryID = c.categoryID AND (c.username =  ? or
userID=-1)
```

Authorization User ID Format: Authentication attribute used in the active identity plug-in. To find this value, go to the Global Settings - Identity Management Setup page in Oracle SES. Enter the value of the Authentication Attribute under the Active Plug-in (for example, `nickname` or `username` or something else).

If you are using the Oracle E-Business Suite R12 identity plug-in, then leave the this parameter blank.

f. Click **Create** to complete the source creation.

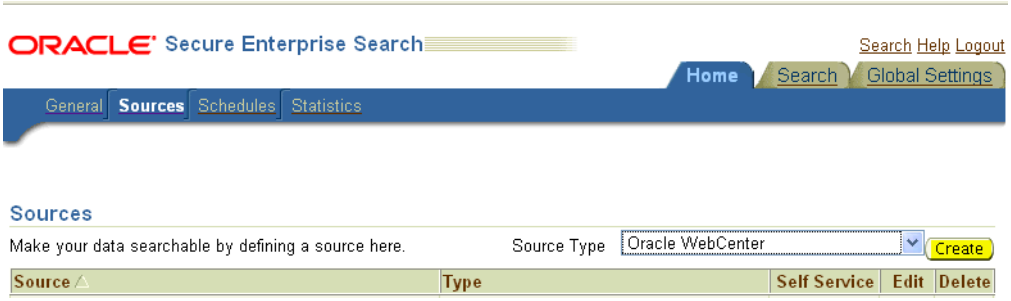
18.5.4.4 Setting Up Oracle SES to Search Portals, Lists, People, and Page Metadata

This section describes how to create the Oracle WebCenter source.

> **See Also:** Section 18.5.6, "Configuring Search Crawlers Using WLST" for an alternative way to create the Oracle WebCenter source

1. Go to the **Home > Sources** page.
2. From the **Source Type** dropdown list, select the **Oracle WebCenter** source type, and click **Create** (Figure 18–25).

Figure 18–25 Create Oracle WebCenter Source



3. Enter the following source parameters:

Note: If WebCenter Portal is fronted with an Oracle HTTP Server, then the Configuration URL used in this step requires the following in `mod_wl_ohs.conf` file.

In a non-clustered environment:

```
<Location /rsscrawl>
SetHandler weblogic-handler
WebLogicHost host_name
WeblogicPort port
</Location>

<Location /sesUserAuth>
SetHandler weblogic-handler
WebLogicHost host_name
WeblogicPort port
</Location>
```

In a clustered environment:

```
<Location /rsscrawl>
WebLogicCluster host_name1:port,host_name2:port
SetHandler weblogic-handler
</Location>

<Location /sesUserAuth>
WebLogicCluster host_name1:port,host_name2:port
SetHandler weblogic-handler
</Location>
```

where *host_name1* and *host_name2* are the cluster nodes, and *port* is the listening port number of the managed server on which the WebCenter Portal application is deployed.

See Also: Section 33.2.3.1, "Installing and Configuring OAM 11g" for detailed information about using WebCenter Portal with Oracle Access Manager

Source Name: *unique_name*

Configuration URL: *host:port_of_WebCenterPortal*`/rsscrawl`; for example, `http://myhost:8888/rsscrawl`

Authentication Type: `BASIC`

User ID: Crawl admin user you registered in Section 18.3.2, "Oracle SES - Configuration"; for example, `mycrawladmin`

Password: Password for the crawl admin user

Realm: `jazn.com`

Oracle SSO Login URL: Leave this field blank.

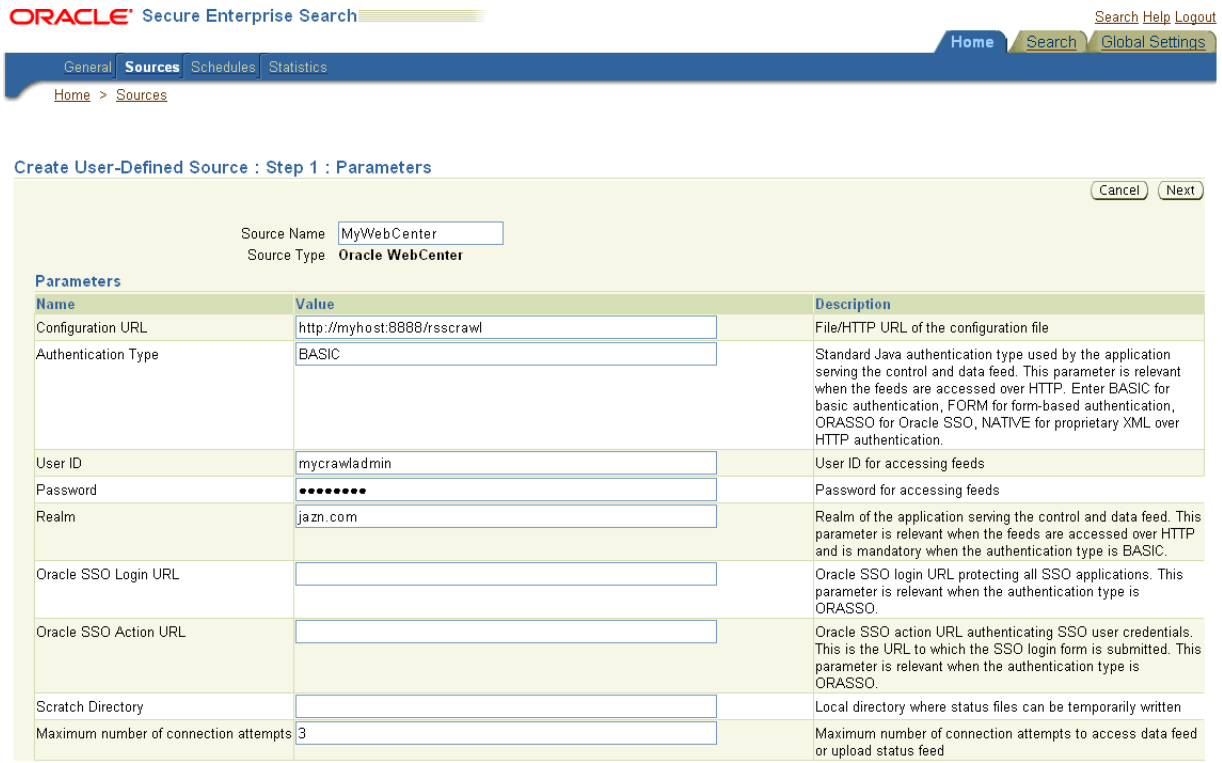
Oracle SSO Action URL: Leave this field blank.

Scratch Directory: Optional. Specify a directory on the system under which the Oracle SES instance resides.

Number of connection attempts: Maximum number of connection attempts to access data feed or upload status feed.

Click **Next** (Figure 18–26).

Figure 18–26 Oracle WebCenter Source Parameters



4. On the Create User-Defined Source : Step 2 : Authorization page, the **Plug-in Class Name** and **Authorization Endpoint** are prepopulated on the page. The **Plug-in Class** name should be `oracle.webcenter.search.auth.plugin.WebCenterAuthManager`.

   Enter the following plug-in parameters:

Jar File Name: `webcenter/search-auth-plugin.jar` (Note: This must be changed from the default value.)

Realm: `jazn.com`

User ID: Crawl admin user you registered Section 18.3.2, "Oracle SES - Configuration"; for example, `mycrawladmin`

Password: Password for the crawl admin user

Authorization User ID Format: Authentication attribute used in the active identity plug-in. To find this value, go to the Global Settings - Identity Management Setup page in Oracle SES. Enter the value of the Authentication Attribute under the Active Plug-in (for example, `nickname` or `username` or something else). If you are using the Oracle E-Business Suite R12 identity plug-in, then leave the this parameter blank.

5. Click **Create** to complete the source creation.

18.5.4.5 Excluding Components from the Spaces Crawler

The spaces crawler collects data for searching the following components:

- `oracle.webcenter.peopleconnections.profile` (people)
- `oracle.webcenter.community` (portals)
- `oracle.webcenter.page` (page metadata)
- `oracle.webcenter.list` (lists)

Use the URL parameter `?excludedServiceIds` to disable search for any of these components. That is, in the Oracle SES administration tool, on the Home - Sources page for the Oracle WebCenter source, the `?excludedServiceIds` in the **Configuration URL** parameter should equal to the comma-delimited list of service IDs to exclude.

Example 18–5 Disable Crawling of People Connections Profiles

```
http://host:port/rsscrawl?excludedServiceIds=oracle.webcenter.peopleconnections.profile
```

Example 18–6 Disable Crawling of Page Metadata

```
http://host:port/rsscrawl?excludedServiceIds=oracle.webcenter.page
```

Example 18–7 Disable Crawling of Profiles and Page Metadata

```
http://host:port/rsscrawl?excludedServiceIds=oracle.webcenter.peopleconnections.profile,oracle.webcenter.page
```

18.5.4.6 Additional Oracle SES Configuration

This section describes the required steps in the Oracle SES administration tool.

1. Create a *source group* that includes the names of the Content Server, Discussions, Announcements, and WebCenter Portal sources you created.
   a. Go to the Search - Source Groups page, and click **Create**.
   b. Enter the same source group name entered in Section 18.5.1, "Setting Up WebCenter Portal for Oracle SES."
   c. From the **Select Source Type** dropdown list, select each source type (Database, Oracle Content Server, Oracle WebCenter), and then from the Available

Sources listed for each source type, move the source you created for that source type into the Assigned Sources list.

d. Click **Finish**.

2. Optionally configure the security filter lifespan. This refreshes the authorization policies for users in the system. It is best to have a short lifespan when user policies change frequently. (This chapter uses Oracle Internet Directory identity plug-in as the example.)

   For example, on the Global Settings - Query Configuration page, under **Secure Search Configuration**, enter `0` for **Security Filter Lifespan (minutes)**.

   Valid values for the security filter lifespan are between 0 minutes (no cache) and 526500 minutes (cache for one year).

3. To index everything, you must force a full crawl for each source; that is, you must change the existing incremental crawl schedule for each source to first process ALL documents.

   This step is very important, in that searching does not work unless the content is first indexed completely.

   Note: You can set the schedule for the spaces crawler with the **fullCrawlIntervalInHours** parameter in WLST or the **Full Crawl Interval** parameter in Fusion Middleware Control.

   Go to the Home - Schedules page, select the source schedule, and click **Edit** to force a full crawl.

   After each source has been crawled, go back to the same page and change the crawl policy back to incremental (index documents that have changed since the previous crawl). Also, in the Frequency section of the page, select a non-manual type for running incremental crawl (for example, weekly or daily).

   Note: Before the first crawl of the Content Server, remember to go to the Content Server Administration page, select **SES Crawler Export**, and take a snapshot. For more information, see Section 18.5.2, "Setting Up Oracle WebCenter Content Server for Oracle SES."

18.5.4.7 Configuring Oracle SES Facets and Sorting Attributes

Facets are Oracle SES objects that let users refine searches by navigating indexed data without running a new search. You must first define facets (using the provided files) in Oracle SES. Facets defined in Oracle SES are picked up in WebCenter Portal though the **Tools and Services - Search** administration page.

Reminder: **Oracle SES 11.2.2.2** supports faceted search with WebCenter Portal, but earlier releases of Oracle SES search do not.

WebCenter Portal provides the following input files to the Oracle SES Admin API command line interface:

- `facet.xml`: This configures facets in Oracle SES.
- `searchAttrSortable.xml`: This defines attributes for absolute sort.

Locate these files in `oracle.webcenter.framework/ses/webcenter_portal_ses_admin.zip`. Unzip this file, and follow the instructions in the `readme.txt` file.

Running these two files from Oracle SES creates the following facets:

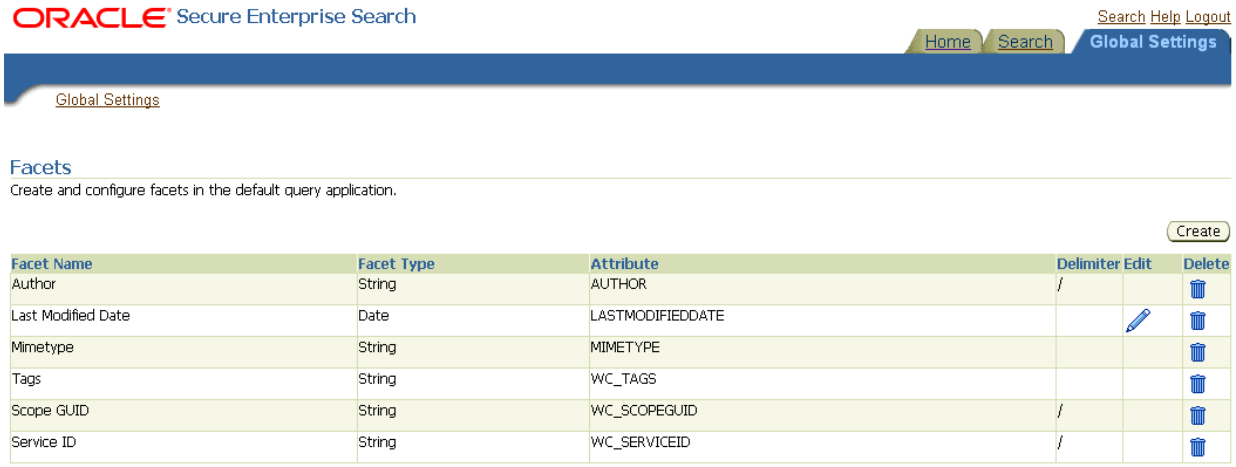
- `Author`
- `Last Modified Date`
- `Mimetype`
- `Tags`
- `Scope GUID` (This appears as the **Portal** facet. This value is converted to the portal display name in the search results page.)
- `Service ID` (This facet does not appear in the user interface. All enabled tools and services display in the search results page.)

Notes: The `facet.xml` and `searchAttrSortable.xml` scripts are mandatory. Creating facets in Oracle SES alone is not sufficient for search in WebCenter Portal.

Additionally, the `Scope GUID` and the `Service ID` facets are mandatory. Facet names are case-sensitive. You must have these exact facet names.

After you run these files, you can view facets in the Oracle SES administration tool on the Global Settings - Facets page (Figure 18–27).

Figure 18–27 Oracle SES Facets



To create a new facet, on the Global Settings - Facets page, click **Create**. Enter a name for the facet and the search attribute from which the facet value should be generated. For String facet types, you must also enter the path delimiter. This is a single character used for demarcation for displaying the facet tree hierarchy for the selected facet tree node on the query page, for example, "tools/power tool/drills", where "/" is the path delimiter. You can set it to blank if the facet tree is one-level deep; that is, its nodes do not have child nodes.

Click **Create and Customize** to create a facet and configure its nodes on the Edit Facet page. You can configure facet nodes for a facet of Date type or Number type. For

example, for the Last Modified Date facet, you can create nodes like Last Year, Last Month, Today, Between two specific days, and so on.

The Node Configuration tab displays a facet hierarchy in tree format as well as in XML format, where you can add, edit, and delete child nodes for the selected facet node. After editing the facet nodes, click **Apply** to save the changes.

Note: Do not modify or delete the `Scope GUID` or `Service ID` facets.

Changes you make in Oracle SES are picked up in WebCenter Portal when the application specialist goes to the **Tools and Services - Search** administration page. WebCenter Portal does not detect changes to facets until this Search administration page is opened. WebCenter Portal remembers the facets selected for use by each portal.

18.5.5 Configuring Oracle SES Version Using WLST

You must run the `setSESVersion` WLST command to obtain and store version information for the Oracle SES instance associated with your default connection. This command enables faceted search and the Tools and Services - Search administration page, which is necessary for customizing search settings with Oracle SES 11.2.2.2. To confirm that the Oracle SES version is set correctly, run the `listSESVersion` WLST command.

Example 18–8 shows these commands. For full command syntax and examples, see the "setSESVersion" and "listSESVersion" sections in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

Example 18–8 Enable Facet Query and the Tools and Services Search Admin Page

```
setSESVersion(appName='webcenter',
 sesUrl='http://myhost.com:5720/search/api/admin/AdminService',
 sesSchema='searchsys', sesPassword='password'
 listSESVersion(appName='webcenter',
 sesUrl='http://myhost.com:5720/search/api/admin/AdminService')
```

18.5.6 Configuring Search Crawlers Using WLST

You can use WLST commands to create crawlers and to start, stop and delete crawler schedules. These commands let you crawl new data in Oracle SES or delete old crawlers if the configuration data changes.

The following examples show some of these commands. For more information, see the "Search - Oracle SES Search Crawlers" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

Example 18–9 Create a Spaces Crawler in WLST

```
createSpacesCrawler(
appName='webcenter', host='myWebcenterHost', port='myWebcenterPort',
sesUrl='http://mySEShost.com:7777/search/api/admin/AdminService',
sesPassword='mySESAdminPassword', crawlUser='webcenter-crawl-user',
crawlPassword='webcenter-crawl-user-pw', scratchDir='/tmp',
authUserIdFormat='authentication-id-format', crawlingMode='ACCEPT_ALL',
recrawlPolicy='PROCESS_ALL', freqType='MANUAL', startHour=1,
hoursBetweenLaunches=1, startDayOfWeek='MONDAY', startDayOfMonth=1,
daysBetweenLaunches=1, weeksBetweenLaunches=1, monthsBetweenLaunches=1,
sesSchema='eqsys')
```

Example 18–10 Create a Documents Crawler in WLST

```
createDocumentsCrawler(
appName='webcenter', host='myWebcenterHost', port='myWebcenterPort',
sesUrl='http://mySEShost.com:7777/search/api/admin/AdminService',
sesPassword='mySESAdminPassword',
configUrl='http://myContentServerHost:myContentServerPort/cs/idcplg?IdcService=SES
_CRAWLER_DOWNLOAD_CONFIG&source=default',
user='ContentServer_crawl_user', password='ContentServerCrawlPassword',
scratchDir='/tmp',
httpEndpoint='http://myContentServerHost:myContentServerPort/cs/idcplg',
displayUrl='http://myContentServerHost:myContentServerPort/cs', realm='Idc
Security /cs/idcplg',
authUserIdFormat='authentication-id-format', pipelineName='Document-pipeline',
crawlingMode='ACCEPT_ALL', recrawlPolicy='PROCESS_CHANGED', freqType='MANUAL',
startHour=1, hoursBetweenLaunches=1, startDayOfWeek='MONDAY', startDayOfMonth=1,
daysBetweenLaunches=1, weeksBetweenLaunches=1, monthsBetweenLaunches=1,
sesSchema='eqsys')
```

Example 18–11 Create a Discussions Crawler in WLST

```
createDiscussionsCrawler(
appName='webcenter', host='myWebcenterHost', port='myWebcenterPort',
sesUrl='http://mySEShost.com:7777/search/api/admin/AdminService',
sesPassword='mySESAdminPassword',
dbConnString='jdbc:oracle:thin:@database-host:database-port:database-sid',
user='Jive-crawler-schema', password='Jive-crawler-schema-pw',
authUserIdFormat='authentication-id-format', crawlingMode='ACCEPT_ALL',
recrawlPolicy='PROCESS_ALL', freqType='MANUAL', startHour=1,
hoursBetweenLaunches=1, startDayOfWeek='MONDAY', startDayOfMonth=1,
daysBetweenLaunches=1, weeksBetweenLaunches=1, monthsBetweenLaunches=1,
sesSchema='eqsys')
```

Notes:

- For *authentication-id-format*, use 'nickname' if the Identity Management plug-in on Oracle SES is set to Oracle Internet Directory; otherwise, use the value of the **Authentication Attribute** parameter on the Identity Management plug-in on Oracle SES.
- For *database-host*, use Oracle WebCenter Portal's Discussion Server database host name
- For *Jive-crawler-schema*, use the Discussions server crawler schema name. Determine the prefix from RCU, and use *rcu-prefix*_DISCUSSION_CRAWLER.
- For sesSchema, the default value is searchsys, which is the default admin user name for Oracle SES 11.2.2.2; however, a different name may have been specified during installation. If you are using Oracle SES 11.1.*, then set this parameter to eqsys.
- To effect WLST changes, you must restart the managed server on which the application is deployed (by default, WC_Spaces). For more information, see the "Starting and Stopping Managed Servers Using WLST" section in *Oracle Fusion Middleware Administrator's Guide*.

18.5.7 Tips for Crawling Page Metadata

To crawl page metadata in a Portal Framework application, follow these guidelines:

- In page templates, render pages as links using go links (`af:goLink`) instead of command links (`af:commandLink`).
- Disable iterative development in JDeveloper during crawling. Iterative development lets you make changes to your application while it is running and immediately see the effect of those changes by refreshing the page in your browser. The iterative development feature works by disabling certain optimization features.

  Iterative development is enabled by default. To turn it off:

  1. In JDeveloper, from the Application menu, select **Application Properties**.
  2. Along the left side of the Application Properties dialog, expand the **Run** node.
  3. Select **WebCenter Portal**.
  4. Deselect **Enable Iterative Development**.
  5. Click **OK**.

18.6 Configuring Oracle SES to Search Portal Framework Applications

With WebCenter Portal Framework applications, Oracle SES is set as the default and preferred search platform. Configuring Oracle SES to search Portal Framework applications requires similar steps, but Portal Framework applications do not support the spaces crawler.

This section describes the steps to set up Oracle SES to search Portal Framework applications:

- Section 18.6.1, "Setting Up Oracle WebCenter Content Server for Oracle SES"
- Section 18.6.2, "Setting Up Oracle WebCenter Portal's Discussion Server for Oracle SES"
- Section 18.6.3, "Setting Up Oracle SES to Search WebCenter Portal Framework"
- Section 18.6.4, "Setting Up WebCenter Portal Framework Applications for Oracle SES"

See Also: Section 18.1, "About Search with Oracle SES"

Note: For an overview of the tasks that must be performed to enable Oracle SES as the search engine in Portal Framework applications, see Section 18.2, "Configuration Roadmaps for Oracle SES in WebCenter Portal." There may be various acceptable ways and orders to perform the required tasks.

18.6.1 Setting Up Oracle WebCenter Content Server for Oracle SES

This section describes how to configure Oracle WebCenter Content Server to be crawlable by Oracle SES (in particular, the Content Server that your Portal Framework application uses for storing documents).

The following steps must be done from within the Content Server.

See Also: Content Server online help for information on administering roles and users in Content Server

1. Create a crawl user.

   If you want users with the `admin` role to crawl, then use an admin user account as the crawl user.

   If you want non-admin users to crawl, then follow these steps:

   a. Create the role `sescrawlerrole`.

   b. Create the user `sescrawler`, and assign it the `sescrawlerrole` role. This user creates the Content Server source in Oracle SES.

   c. Add `sceCrawlerRole=sescrawlerrole` to `config.cfg` (located in *`MW_HOME`*`/user_projects/domains/`*`yourdomain`*`/ucm/cs/config`).

      Alternatively, you can append the `sceCrawlerRole=sescrawlerrole` line in the WebCenter Content Server user interface (Administration - General Configuration - Additional Configuration Variables).

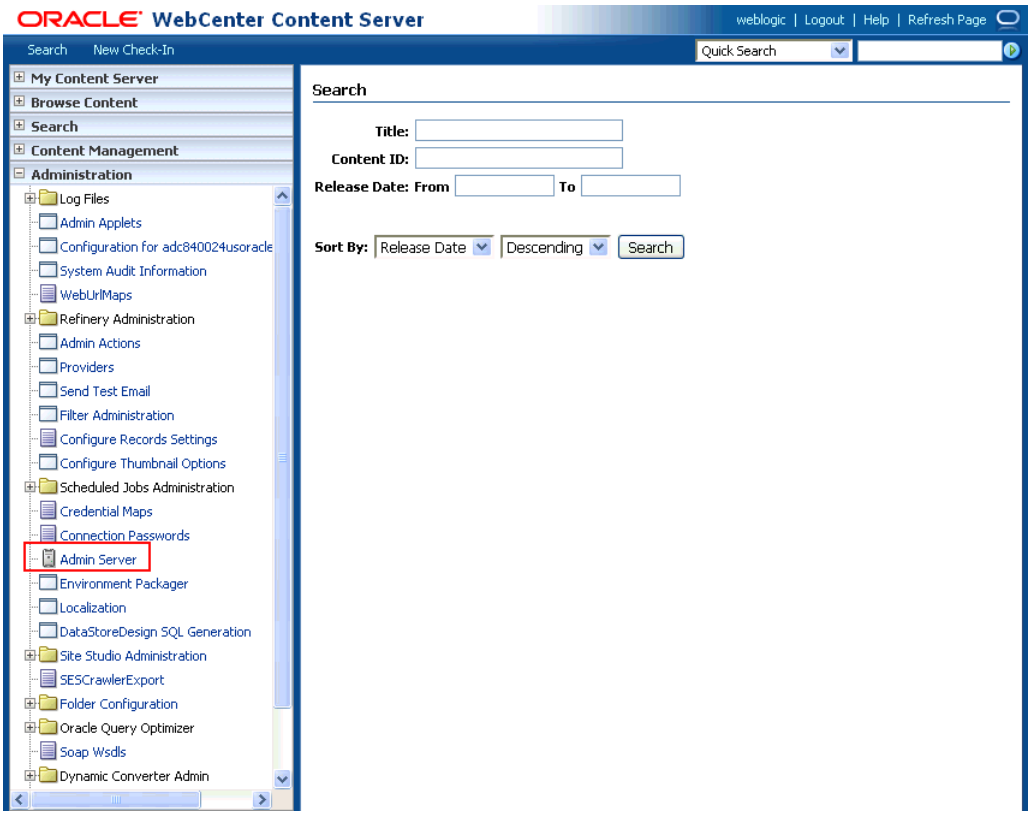
2. Restart the Content Server.

3. In the Content Server console, install the SESCrawlerExport component on the content server, if not done:

   a. Log on to the Content Server as a system administrator. For example: `http://`*`host:port`*`/cs`.

   b. From the Administration dropdown menu, select **Admin Server** (Figure 18–28).

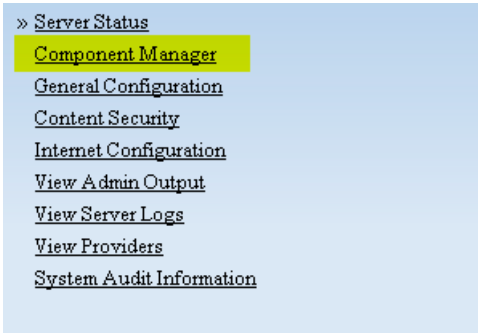
Figure 18–28 Content Server Administration



c. Click the button with the instance name.

d. Click **Component Manager** from the menu list on the left pane (Figure 18–29).

Figure 18–29 WebCenter Content Server Component Manager



e. Select **SESCrawlerExport** under Integration and click **Update**.

f. Enter configuration parameters. (You can change configuration parameters after installation.)

Disable security on authentication and authorization APIs provided by the SESCrawlerExport; that is, set **Disable Secure APIs** to `false`. This lets security provided by the SESCrawlerExport be done internally instead of by the Content Server.

Additionally, in clustered environments only, the **feedLoc** parameter must specify a location on the shared disk accessed by the nodes of the Content

Server, and they each must reference it the same way; for example, `sharedDrive/dir1/dir2`. Note that this is not the default location (relative path) provided.

g. Restart the Content Server.

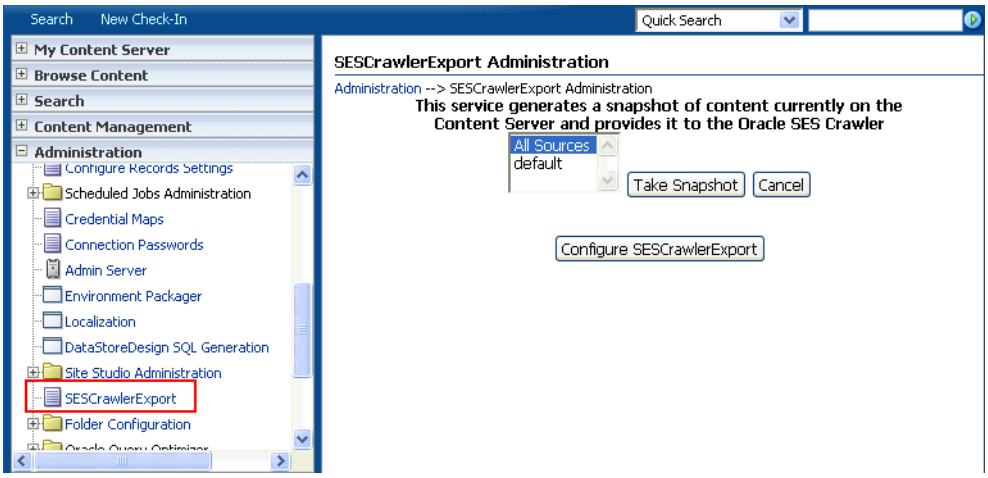
4. Take a snapshot of the Content Server repository.

   a. Log on to the Content Server as a system administrator. For example: `http://host:port/cs`.

   b. From the Administration dropdown menu, select **SESCrawlerExport**.

   c. Select **All sources**, and click **Take Snapshot** (Figure 18–30).

Figure 18–30 Content Server Snapshot



It is important to take a snapshot before the first crawl or any subsequent full crawl of the source.

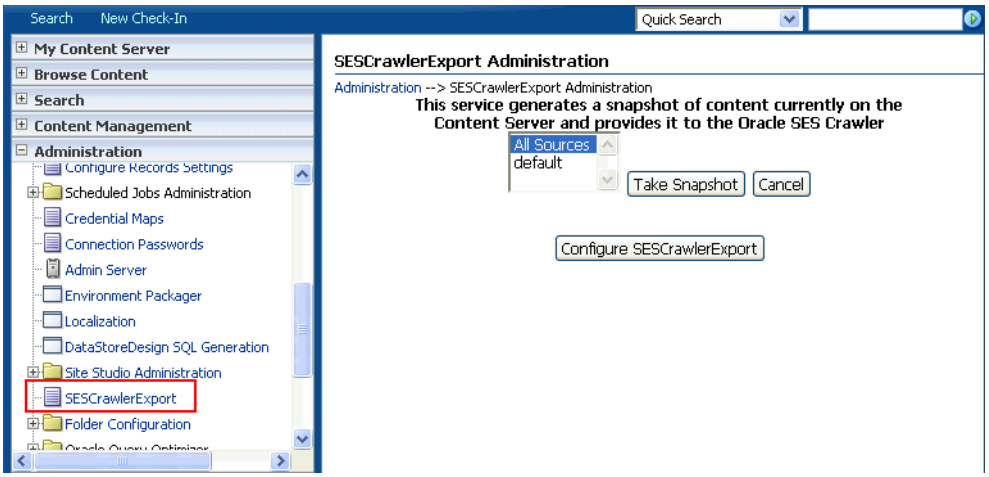
The snapshot generates `configFile.xml` at the location specified during component installation, and feeds are created at the subdirectory with the source name under **feedLoc**.

5. If the Content Server is configured for web rendition, then items in the Content Server are rendered in PDF format. The content item's native MIME type rendition is overwritten. For example, the MIME type of a Microsoft Office Word document is 'application/msword', but when the Content Server uses web rendition the MIME type becomes 'application/pdf'. A search query with the `Mimetype` parameter set to 'application/msword' does not return Word documents.

   If your Content Server is configured to use web rendition, then you must configure the Content Server metatdata list to include the `dFormat` value, so that required MIME types are exported to Oracle SES. This is necessary to be able to narrow searches by MIME type.

   a. Back on the SESCrawlerExport Administration page, click **Configure SESCrawlerExport** (Figure 18–31).

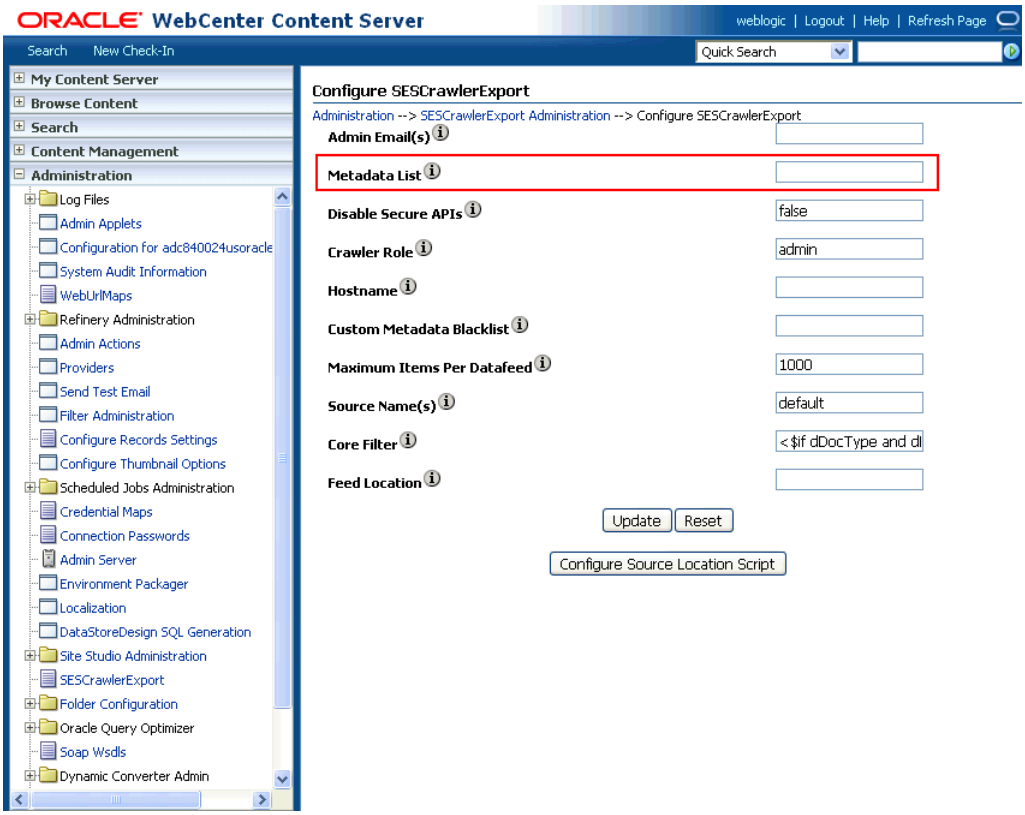
Figure 18–31 Content Server Snapshot



b. By default, the **Metadata List** field is blank (Figure 18–32). Optionally, add to this field any custom metadata values you require (beginning with `x`). For example, the following entry for **Metadata List** includes custom attributes:

```
xCollectionID,xWCTags,xRegionDefinition
```

Figure 18–32 Content Server Metadata List



c. If you are using a version of Content Server *earlier than* 11.1.1.8, then you must add the `dFormat` value to this **Metadata List** field. When the blank default value is changed, the default values are removed, and they must be added

back. (Content Server 11.1.1.8 includes most values required for search, including `dFormat`.)

Therefore, if you are using a version of Content Server earlier than 11.1.1.8, you must manually enter the default value for **Metadata List** as follows (plus any custom metadata fields beginning with 'x'):

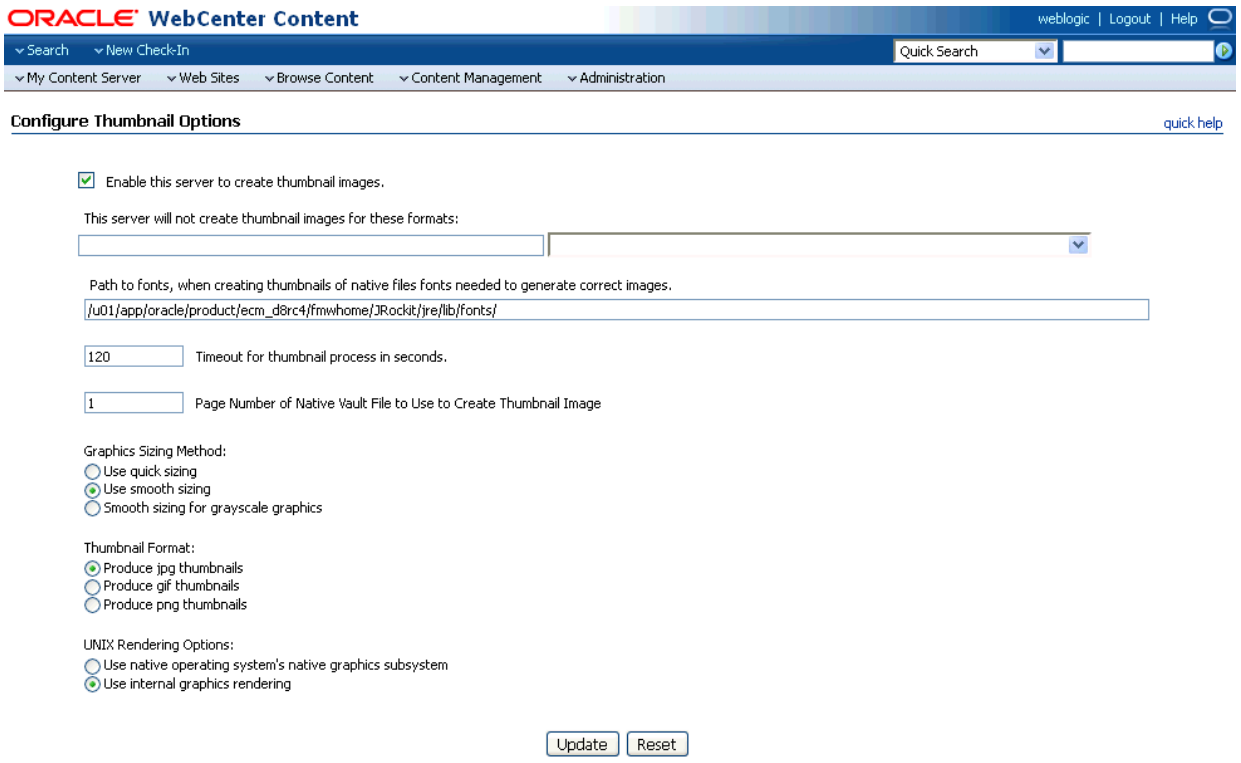
```
dFormat,dID,dDocName,dRevLabel,dDocType,dDocAccount,dSecurityGroup,
dOriginalName,dReleaseDate,dOutDate,dDocCreator,dDocLastModifier,dDocCreate
dDate,dDocFunction
```

6. Configure Thumbnail Options for faceted search.

Note: Oracle SES 11.2.2.2 supports document thumbnails, while earlier releases of Oracle SES do not.

On the **Administration** tab, select **Configure Thumbnail Options** to enable document thumbnails in search results. Leave the default settings as is, and click **Update** (Figure 18–33).

Figure 18–33 Configure Thumbnail Options



See Also: The `Deployment Guide.pdf` included with the product for detailed information on Content Server configuration

18.6.2 Setting Up Oracle WebCenter Portal's Discussion Server for Oracle SES

This section describes how to configure Oracle WebCenter Portal's Discussion Server to be crawlable by Oracle SES (in particular, the discussions server that your Portal Framework application uses for storing discussions and announcements).

Note: These steps are not required if you have a new installation of WebCenter Portal (with an Oracle database) and Oracle WebCenter Portal's Discussion Server. They are only required if you are using upgraded (patched) instances.

You can find database schema details for the corresponding data sources from your Oracle WebLogic Server console.

---

1. Run the Repository Creation Utility (RCU) to confirm that the discussions crawler WebCenter Portal component has been installed on the system.

   - Oracle databases:

     Verify that the Oracle WebCenter Portal's Discussion Server back end has been configured properly by noting that the *MyPrefix*_DISCUSSIONS user is installed in RCU.

     Then verify that the discussions crawler has been configured properly by noting that the *MyPrefix*_DISCUSSIONS_CRAWLER user is installed in RCU.

   - Microsoft SQL Server databases:

     Verify that the Oracle WebCenter Portal's Discussion Server back end has been configured properly by noting that the *MyPrefix*_DISCUSSIONS user is installed in RCU.

     Then verify that the discussions crawler has been configured properly by noting that the *MyPrefix*_DISCUSSIONS_CRAWLER user is installed in RCU.

   - IBM DB2 databases:

     Verify that the Oracle WebCenter Portal's Discussion Server back end has been configured properly by noting that the *MyPrefix*_DS user is installed in RCU.

     Then verify that the discussions crawler has been configured properly by noting that the *MyPrefix*_DC user is installed in RCU.

     Note: For IBM DB2 databases, *MyPrefix* is limited to five characters. IBM DB2 uses operating system users for authentication (that is, the database user is actually an operating system user). Because some operating systems have an eight character limit for user names, this requires an eight character user name.

   If the discussions crawler component is not installed, then you must install it using RCU, selecting the same prefix that was used for the Oracle WebCenter Portal's Discussion Server component. Also, during the tablespace specification step in RCU, select *Prefix*_IAS_DISCUSSIONS as the default tablespace. This installs the user for Oracle SES.

   For more information, see Chapter 42, "Deploying Portal Framework Applications."

2. For instances upgraded from WebCenter 11.1.1.1.0, run the following tool to upgrade the data in the Oracle WebCenter Portal's Discussion Server database schema, if you have not run the tool yet:

Note: This step is necessary only if the instance is upgraded from WebCenter Portal 11.1.1.1.0. For instances installed after WebCenter Portal 11.1.1.1.0, this is not required.

```
java -jar \
$MW_HOME/discussionserver/discussionserver-upgradeforses.jar \
<command_line_parameters>
```

where *command_line_parameters* are the following MDS schema details and discussions database schema details:

```
-mds_jdbc_user user_id \
-mds_jdbc_password password \
-mds_jdbc_url url \
-discussions_jdbc_user user_id \
-discussions_jdbc_password password \
-discussions_jdbc_url url
```

where *mds_jdbc_user*, *mds_jdbc_password*, and *mds_jdbc_url* are the values to log in to the MDS schema, and *discussions_jdbc_user*, *discussions_jdbc_password*, and *discussions_jdbc_url* are the values to log in to the discussions database schema.

For example:

```
java -jar
$MW_HOME/as11r1wc/discussionserver/discussionserver-upgradeforses.jar\
-mds_jdbc_user foo \
-mds_jdbc_password myPassword1 \
-mds_jdbc_url jdbc:oracle:thin:@host:port:SID \
-discussions_jdbc_user foo \
-discussions_jdbc_password myPassword1 \
-discussions_jdbc_url jdbc:oracle:thin:@host:port:SID
```

18.6.3 Setting Up Oracle SES to Search WebCenter Portal Framework

The steps in this section must be performed in the Oracle SES administration tool.

The following steps are required:

1. Section 18.6.3.1, "Logging on to the Oracle SES Administration Tool"
2. Section 18.6.3.2, "Setting Up Oracle SES to Search Documents"
3. Section 18.6.3.3, "Setting Up Oracle SES to Search Discussions and Announcements"
4. Section 18.6.3.4, "Configuring Oracle SES Facets and Sorting Attributes"
5. Section 18.6.3.5, "Additional Oracle SES Configuration"

See Also: Confirm that you have installed all required patches for Oracle SES. For the latest information on required patches, see the "Back-End Requirements for Search" section in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal* and the Release Notes.

For detailed information about Oracle SES configuration, see the Oracle SES documentation included with the product. (This is listed in the WebCenter Portal product area on the Oracle Fusion Middleware documentation library.)

18.6.3.1 Logging on to the Oracle SES Administration Tool

Open the Oracle SES administration tool:

1. Open a browser and enter the URL provided after the Oracle SES installation. (This has the form `http://host:port/search/admin/index.jsp`.)
2. Log on with the Oracle SES admin user name `eqsys` and the password specified during installation.
   - For **Oracle SES 11.2.2.2**, the default admin user name is `searchsys`; however, a different name may be specified during installation.
   - For **Oracle SES 11.1.***, the admin user name is `eqsys`.

18.6.3.2 Setting Up Oracle SES to Search Documents

To search documents using Oracle SES, you must first set up a Document Service Manager (with a Document Service Instance and a Document Service Pipeline), and then create a Content Server source.

1. Configure the Document Service Manager (one time for each Oracle SES instance).

   Note: Document services are plug-ins involved in the processing of a document when it is being crawled. A document service allows your Portal Framework application to add indexable attributes for documents used in the application.

   a. On the Global Settings - Document Services page, click **Create**. Select **Create New Manager**, click **Next**, and enter the following parameters:

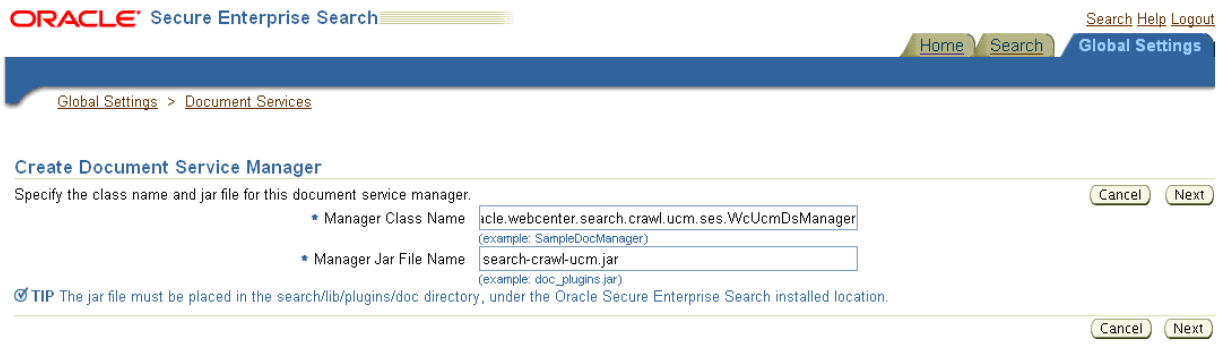
      Manager Class Name:
      `oracle.webcenter.search.crawl.ucm.ses.WcUcmDsManager`

      Manager Jar File Name: `search-crawl-ucm.jar`

      Note: The `webcenter_doc_pipeline_plugin.zip` file installs `Oracle_Home/search/lib/plugins/doc/search-crawl-ucm.jar`.

      Click **Next**, and then click **Finish** (Figure 18–34).

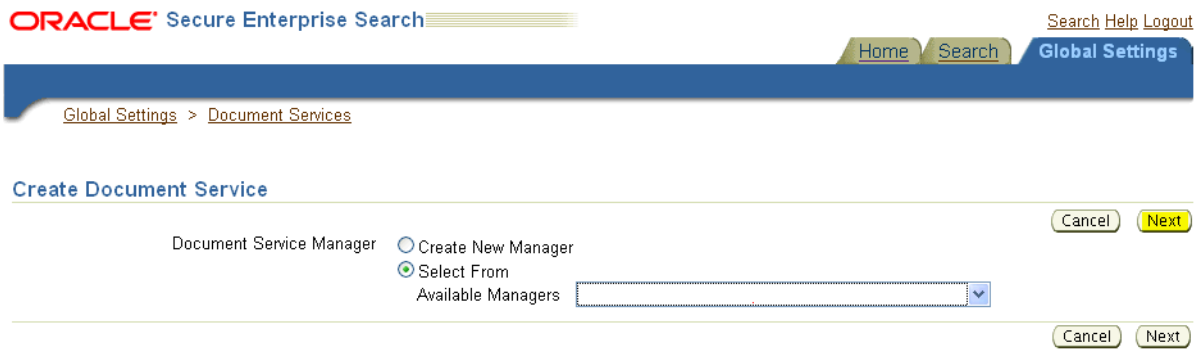
Figure 18–34 Creating a Document Service Manager in Oracle SES



b. Create the Document Service Instance.

Again, on the Global Settings - Document Services page, click **Create.** This time, select **Select From Available Managers** with **Secure Enterprise Search WebCenter UCM Plugin**, and click **Next** (Figure 18–35).

Figure 18–35 Create Document Service



Enter the following parameters:

Instance Name: Enter any name here to be used while creating the document pipeline.

WebCenter Application Name: This must be left blank.

Connection Name: The Content Server connection name in your Portal Framework application.

WebCenter URL Prefix: The host and port where the application is deployed, plus the context root; for example: `http://myhost:8888/myPortalApp`, where `myPortalApp` is the context root of the application.

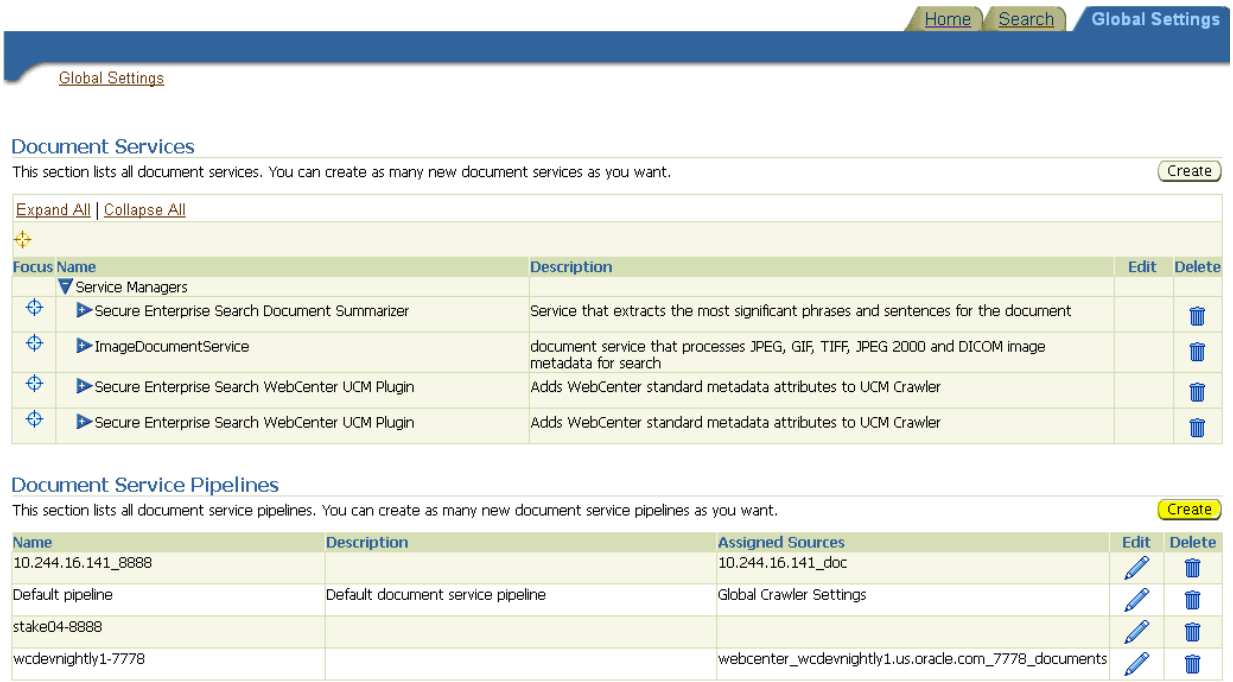
Note: Use Fusion Middleware Control or the `listDocumentsSpacesProperties` command to determine the application name and connection name, as described in Section 18.5.1, "Setting Up WebCenter Portal for Oracle SES."

---

c. Create the Document Service Pipeline. This invokes the document service instance.

Again, on the Global Settings - Document Services page, under the **Document Services Pipelines** section, click **Create** (Figure 18–36).

Figure 18–36 Creating the Document Service Pipeline



d. On the Create Document Service Pipeline page, enter any custom name for this pipeline. The document service instance you created in the previous step should be listed under **Available Services**. Select that document service instance, and use the arrow button to move it under **Used in pipeline**.

2. Create the Content Server source for documents.

a. Go to **Home > Sources**.

b. From the Source Type dropdown list, select **Oracle Content Server**, and click **Create** (Figure 18–37).

Figure 18–37 Create Oracle Content Server Source

c. Enter the following parameters:

Source Name: *unique_name*

Configuration URL: *Content_Server_SES_Crawler_Export_endpoint*; for example,

`http://host:port/cs/idcplg?IdcService=SES_CRAWLER_DOWNLOAD_CONFIG&source=default`

Note: The `source=default` parameter denotes the name of the source created in the configuration of the SES Crawler Export. The default one is created automatically and called "default."

Authentication Type:

If the Content Server is not protected by SSO, then enter `NATIVE`.

If the Content Server is protected by Oracle SSO, then enter `ORASSO`.

User ID: The user to crawl the Content Server must have the `sceCrawlerRole` role defined. The `sceCrawlerRole` is a configuration parameter in `SESCrawlerExport`. Typically, administrators create a special role, assign it no privileges to view content, then create a user account that has this role.

If you do not set up a specific `sceCrawlerRole`, then admin credentials are required to crawl. The `sysadmin` user ID works by default.

If Authentication Type is `ORASSO`, then enter a user ID (and password) of a user in the identity management server fronted by Oracle SSO. This user must have been granted the same permissions as sysadmin. If it is not possible to grant those permissions, then delete the "remote" user corresponding to this user in the Content Server, and create a "local" version of the user (same name) in the Content Server.

Password: Password for this Content Server user.

Realm:

If Authentication Type is `NATIVE`, then enter `Idc Security /cs/idcplg`, where `/cs/` is the context root you provided when you installing the Content Server.

If Authentication Type is `ORASSO`, then leave this parameter blank.

Scratch Directory: Optional. Specify a directory on the system under which the Oracle SES instance resides.

Oracle SSO Login URL:

If Authentication Type is `ORASSO`, then specify a value for Oracle SSO. For example:
`https://login.oracle.com/mysso/signon.jsp?site2pstoretoken=`

If Authentication Type is `NATIVE`, then leave this field blank.

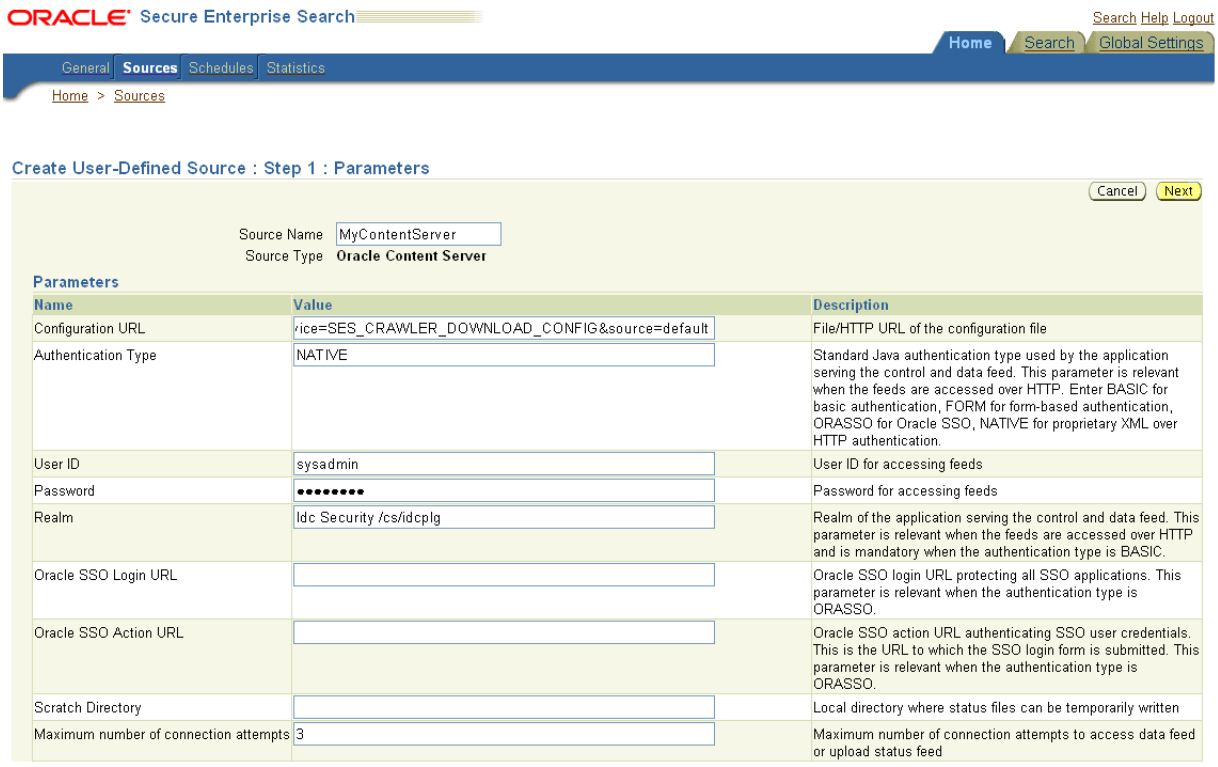
Oracle SSO Action URL:

If Authentication Type is `ORASSO`, then specify a value for Oracle SSO. For example: `https://login.oracle.com/sso/auth`

If Authentication Type is `NATIVE`, then leave this field blank.

Click **Next** (Figure 18–38).

Figure 18–38 Oracle Content Server Source Parameters



d. On the Create User-Defined Source : Step 2 : Authorization page, enter the following parameters in the Authorization Manager section, if not entered by default:

Plug-in Class Name: `oracle.search.plugin.security.auth.stellent.StellentAuthManager`

Jar File Name: `oracleapplications/StellentCrawler.jar`

HTTP endpoint for authorization: for example, `http://host:port/cs/idcplg`

Display URL Prefix: for example, `http://host:port/cs`

Authentication Type: `NATIVE` or `ORASSO`

Administrator User: The user to crawl the Content Server must have the `sceCrawlerRole` role defined. The `sceCrawlerRole` is a configuration parameter in `SESCrawlerExport`. Typically, administrators create a special role, assign it no privileges to view content, then create a user account that has this role.

If you do not set up a specific `sceCrawlerRole`, then admin credentials are required to crawl. The `sysadmin` user ID works by default.

If Authentication Type is `ORASSO`, then enter a user ID (and password) of a user in the identity management server fronted by Oracle SSO. This user must have been granted the same permissions as sysadmin. If it is not possible to grant those permissions, then delete the "remote" user corresponding to this user in the Content Server, and create a "local" version of the user (same name) in the Content Server.

Administrator Password: Password for crawl admin user

Authorization User ID Format: Authentication attribute used in the active identity plug-in. To find this value, go to the Global Settings - Identity Management Setup page in Oracle SES. Enter the value of the Authentication Attribute under the Active Plug-in (for example, `nickname` or `username` or something else). If you are using the Oracle E-Business Suite R12 identity plug-in, then leave the this parameter blank.

Realm:

If Authentication Type is `NATIVE`, then enter "`Idc Security /cs/idcplg`", where `/cs/` is the context root you provided when you installing the Content Server.

In Authentication Type is `ORASSO`, then leave this field blank.

e. Click **Create & Customize** (or edit a created source) to see other source parameters. On the **Crawling Parameters** tab, enter the following crawling parameter: `Document Service Pipeline`.

f. Click **Enable** and select the pipeline you created.

18.6.3.3 Setting Up Oracle SES to Search Discussions and Announcements

To search discussions and announcements using Oracle SES, you must first set up several Oracle SES Database sources: three for discussions and one for announcements. The three discussions sources are for forums, topics in forums, and replies in forums. These separate sources enable users to see search results for forums without also seeing results for all the messages and replies in it.

For example, the discussions sources could have the following:

- source name `GS_Forums` and View of `FORUMCRAWLER_VW`
- source name of `GS_Topics` and View of `THREADCRAWLER_VW`
- source name of `GS_Replies` and View of `MESSAGECRAWLER_VW`

The announcements source could have the source name `GS_Announcements` and a View of `ANNOUNCEMENTS_VW`.

Note: There are slightly different steps for Oracle, Microsoft SQL Server, and IBM DB2 databases.

---

1. Configure the JDBC driver:

   a. To crawl a Microsoft SQL Server or IBM DB2 database, download the appropriate JDBC driver jar files into the *`ORACLE_HOME`*`/search/lib/plugins/oracleapplications` directory in Oracle SES.

Note:

- For Microsoft SQL Server: Copy the Microsoft JDBC driver files `sqljdbc.jar` and `sqljdbc4.jar`.
- For IMB DB2: Copy the IBM driver files `db2jcc.jar` and `db2jcc_license_cu.jar` (obtainable from the IBM DB2 UDB client).

If the JDBC drivers for JRE 1.5 and JRE 1.6 are different (for example: `sqljdbc.jar` works for JRE 1.5 and `sqljdbc4.jar` works for JRE 1.6), then perform the following:

- Download both the driver jars into the *ORACLE_HOME*`/search/lib/plugins/oracleapplications` directory in Oracle SES.

- Add an entry for the JRE 1.6 version (`sqljdbc4.jar` for SQLServer) of the driver jar to the `CLASSPATH` element of *ORACLE_HOME*`/search/config/searchctl.conf`.

- Restart the middle tier.

b. Update the `drivers.properties` file with the following information: *DatabaseName:DriverClassName*.

c. Add the JRE 1.5 JDBC driver jar file name to the classpath in `META-INF/MANIFEST.MF` of `appsjdbc.jar` and `DBCrawler.jar`.

For example, change:

```
Class-Path:  sqljdbc.jar rsscrawler.jar  ../../pluginmessages.jar
```

to

```
Class-Path:  db2jcc.jar sqljdbc.jar rsscrawler.jar
../../pluginmessages.jar
```

and change:

```
Class-Path: appsjdbc.jar
```

to

```
Class-Path: db2jcc.jar appsjdbc.jar
```

For a `key` attribute that is not named `KEY`, change the JDBC driver information in the `drivers.properties` file to specify the key attribute name: *database_name: driver_class_name, key_attribute_name*

For example, for a key attribute named `ID`:

```
oracle : oracle.jdbc.driver.OracleDriver, ID
```

In the crawling query, use *key_attribute_name* as the alias for the key value column name. In this example, `ID` is the alias for `KEYVAL`:

```
SELECT keyval id, content, url, lastmodifieddate, lang FROM sales_only
```

Oracle and SQL Server databases: The following default drivers are used if none is specified in `drivers.properties`:

- Oracle: `oracle.jdbc.driver.OracleDriver`
- SQL Server: `com.microsoft.sqlserver.jdbc.SQLServerDriver`

2. Required for IBM DB2 databases only:

a. Make sure that no crawlers are running that use the database crawler source. In the Oracle SES administration tool, check the crawler progress and status on the Home - Schedules page. (Click **Refresh Status**.)

b. Remake the `appsjdbc.jar` file and the `DBCrawler.jar` file. Ensure that the `META-INF/MANIFEST.MF` was updated correctly; otherwise, the crawler fails with the following error in the crawler log file:

```
EQP-80406: Loading JDBC driver failed
```

c. Modify the `ORACLE_HOME/search/lib/plugins/oracleapplications/drivers.properties` file to include the following line:

```
db2: com.ibm.db2.jcc.DB2Driver
```

d. Include the driver jar (`db2jcc.jar`) to the `CLASSPATH` element of `ORACLE_HOME/search/config/searchctl.conf`. For example:

```
#CLASS PATH
CLASSPATH=ORACLE_HOME/search/webapp/config:ORACLE_HOME/search/webapp/
SESAuthenticator.jar:ORACLE_HOME/search/lib/plugins/commons-plugins-
stubs.jar :ORACLE_HOME /search/lib/plugins/oracleapplications/db2jcc.jar
```

e. Edit `JVM_OPTIONS` in the `$ORACLE_HOME/search/config/searchctl.conf` file to add the system property `"-Doracle.home=ORACLE_HOME/search"`. For example:

```
JVM_OPTIONS= -Djava.awt.headless=true
-Dweblogic.RootDirectory=ORACLE_HOME/search/base_domain
-Doracle.home=ORACLE_HOME/search
```

f. Copy the `$ORACLE_HOME/search/lib/plugins/oracleapplications/pluginmessages.jar` file to the `$ORACLE_HOME/search/lib` directory.

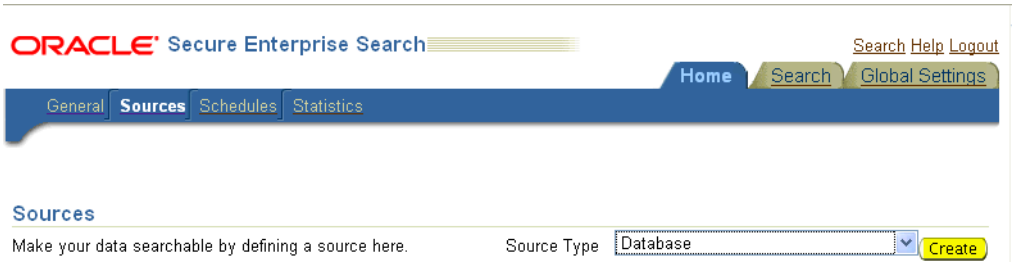
g. Create the database source. Make sure to enter the correct authorization query and confirm that the attribute name used in **Grant Security Attributes** matches the one used in the authorization query; otherwise, users do not get any results when searching for documents.

3. Create the Discussions sources or the Announcements source.

a. In Oracle SES, go to **Home > Sources**.

b. From the Source Type dropdown list, select **Database**, and click **Create** (Figure 18–39).

Figure 18–39 Create Database Source



c. Enter the following parameters:

Source Name: *`unique_name`*; for example, `GS_Forums` to crawl discussion forums (or `GS_Announcements` to crawl announcements)

Database Connection String: Enter one of the following

- Oracle databases: Enter one of the following

`jdbc:oracle:thin:@`*`host:port:sid`*

`jdbc:oracle:thin@`*`host:port/serviceId`*

- IBM DB2 databases: Enter `jdbc:db2://`*`host:port/database_name`*

- Microsoft SQL Server databases: Enter `jdbc:sqlserver://`*`host_or_IP_address:port;database_name`*

User ID: Enter one of the following:

- Oracle databases: The user *`MyPrefix`*`_DISCUSSIONS_CRAWLER` created during Oracle WebCenter Portal's Discussion Server installation

- Microsoft SQL Server databases: The user *`MyPrefix`*`_DISCUSSIONS_CRAWLER` created during Oracle WebCenter Portal's Discussion Server installation

- IBM DB2 databases: The user *`MyPrefix`*`_DC` created during Oracle WebCenter Portal's Discussion Server installation (where *`MyPrefix`* is five characters)

Password: Password for this user

Query: Enter one of the following queries:

```
SELECT * FROM FORUMCRAWLER_VW
SELECT * FROM THREADCRAWLER_VW
SELECT * FROM MESSAGECRAWLER_VW
SELECT * FROM ANNOUNCECRAWLER_VW
```

Use `FORUMCRAWLER_VW` for the source crawling discussion forums.

Use `THREADCRAWLER_VW` for the source crawling topics in discussion forums.

Use `MESSAGECRAWLER_VW` for the source crawling replies in discussion forums.

Use `ANNOUNCECRAWLER_VW` for the source crawling announcements.

URL Prefix: The URL prefix for the Portal Framework application, including host, port, and application name. For example, `http://`*`host:port`*`/myPortalApp`.

Grant Security Attributes: `WCSECATTR`

Note: Previous releases of Content Server used `FORUMID` for **Grant Security Attributes**.

---

d. Click **Next**.

e. On the Create User-Defined Source : Step 2 : Authorization page, enter the following parameters (if not prepopulated) in the Authorization Manager section:

Plug-in Class Name:
`oracle.search.plugin.security.auth.db.DBAuthManager`

Jar File Name: `oracleapplications/DBCrawler.jar`

Authorization Database Connection String: Enter one of the following:

- Oracle databases: Enter one of the following:

`jdbc:oracle:thin:@`*`host:port:sid`*

`jdbc:oracle:thin@`*`host:port/serviceId`*

- IBM DB2 databases: Enter `jdbc:db2://`*`host:port/database_name`*

- Microsoft SQL Server databases: Enter `jdbc:sqlserver://`*`host_or_IP_address:port;database_name`*

User ID: Enter one of the following:

- Oracle databases: Enter the user *`MyPrefix`*`_DISCUSSIONS_CRAWLER`

- Microsoft SQL Server databases: Enter the user *`MyPrefix`*`_DISCUSSIONS_CRAWLER`

- IBM DB2 databases: Enter the user *`MyPrefix`*`_DC` (where *`MyPrefix`* is five characters)

Password: This user password

Single Record Query: `false`

Authorization Query: Enter the following (on one line):

```
SELECT DISTINCT forumID as WCSECATTR
FROM AUTHCRAWLER_FORUM_VW
WHERE username = LOWER(?) UNION SELECT DISTINCT -1 as WCSECATTR
FROM AUTHCRAWLER_FORUM_VW
```

Note: Previous releases of Content Server used the following authorization query:

```
SELECT forumID
FROM AUTHCRAWLER_FORUM_VW
WHERE (username = ? or userID=-1)
UNION SELECT f.forumID
FROM jiveForum f, AUTHCRAWLER_CATEGORY_VW c
WHERE f.categoryID = c.categoryID AND (c.username =  ? or
userID=-1)
```

Authorization User ID Format: Authentication attribute used in the active identity plug-in. To find this value, go to the Global Settings - Identity Management Setup page in Oracle SES. Enter the value of the Authentication Attribute under the Active Plug-in (for example, `nickname` or `username` or something else).

If you are using the Oracle E-Business Suite R12 identity plug-in, then leave the this parameter blank.

f. Click **Create** to complete the source creation.

18.6.3.4 Configuring Oracle SES Facets and Sorting Attributes

Facets are Oracle SES objects that let users refine searches by navigating indexed data without running a new search. You must first define facets (using the provided files) in Oracle SES. Facets defined in Oracle SES are picked up in the Portal Framework though the **Tools and Services - Search** administration page.

Reminder: **Oracle SES 11.2.2.2** supports faceted search with this release of WebCenter Portal. Earlier releases do not.

WebCenter Portal provides the following input files to the Oracle SES Admin API command line interface:

- `facet.xml`: This configures facets in Oracle SES.
- `searchAttrSortable.xml`: This defines attributes for absolute sort.

Locate these files in `oracle.webcenter.framework/ses/webcenter_portal_ses_admin.zip`. Unzip this file, and follow the instructions in the `readme.txt` file.

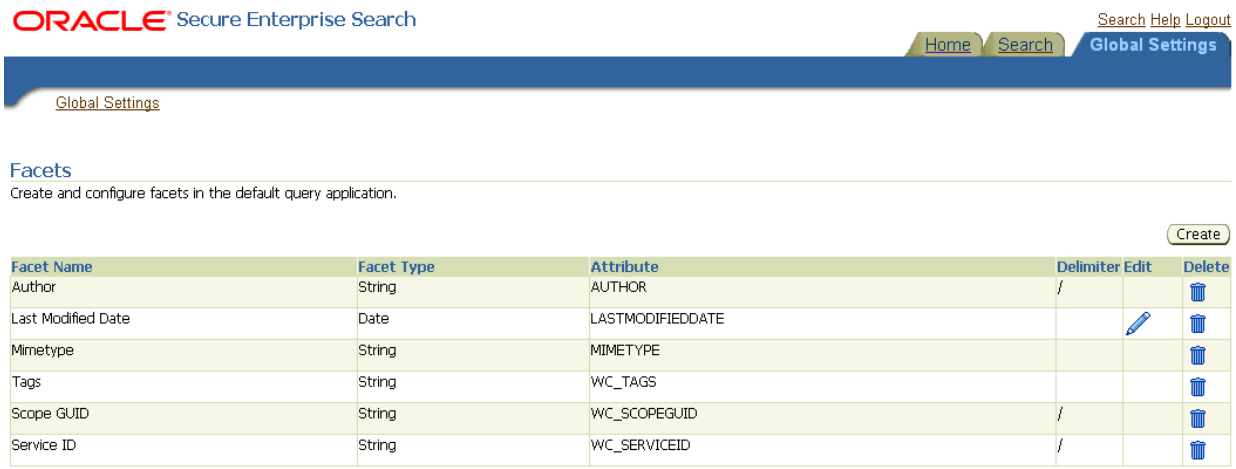
Running these two files from Oracle SES creates the following facets:

- `Author`
- `Last Modified Date`
- `Mimetype`
- `Tags`
- `Scope GUID` (This appears as the **Portal** facet. This value is converted to the portal display name on the search results page.)
- `Service ID` (This facet does not appear in the user interface. All enabled tools and services display on the search results page.)

> **Notes:** The `facet.xml` and `searchAttrSortable.xml` scripts are mandatory. Creating facets in Oracle SES alone is not sufficient for search in your Portal Framework application.
>
> Additionally, the `Scope GUID` and the `Service ID` facets are mandatory. Facet names are case-sensitive. You must have these exact facet names.

After you run these files, you view facets in the Oracle SES administration tool on the Global Settings - Facets page (Figure 18–40).

Figure 18–40 Oracle SES Facets



To create a new facet, on the Global Settings - Facets page, click **Create**. Enter a name for the facet and the search attribute from which the facet value should be generated. For String facet types, you must also enter the path delimiter. This is a single character used for demarcation for displaying the facet tree hierarchy for the selected facet tree

node on the query page, for example, "tools/power tool/drills", where "/" is the path delimiter. You can set it to blank if the facet tree is one-level deep; that is, its nodes do not have child nodes.

Click **Create and Customize** to create a facet and configure its nodes on the Edit Facet page. You can configure facet nodes for a facet of Date type or Number type. For example, for the Last Modified Date facet, you can create nodes like Last Year, Last Month, Today, Between two specific days, and so on.

The Node Configuration tab displays a facet hierarchy in tree format as well as in XML format, where you can add, edit, and delete child nodes for the selected facet node. After editing the facet nodes, click **Apply** to save the changes.

Note: Do not modify or delete the `Scope GUID` or `Service ID` facets.

Changes you make in Oracle SES are picked up in your Portal Framework application when the application specialist goes to the **Tools and Services - Search** administration page. The Portal Framework application does not detect changes to facets until this Search administration page is opened. WebCenter Portal remembers the facets selected for use by each portal.

18.6.3.5 Additional Oracle SES Configuration

This section describes the required steps in the Oracle SES administration tool.

1. Create a *source group* that includes the names of the Content Server, Discussions, Announcements, and WebCenter sources you created.
   a. Go to the Search - Source Groups page, and click **Create**.
   b. Enter the same source group name entered in Section 18.6.4, "Setting Up WebCenter Portal Framework Applications for Oracle SES."
   c. From the **Select Source Type** dropdown list, select each source type (Database, Oracle Content Server, Oracle WebCenter), and then from the Available Sources listed for each source type, move the source you created for that source type into the Assigned Sources list.
   d. Click **Finish**.
2. Optionally configure the security filter lifespan. This refreshes the authorization policies for users in the system. It is best to have a short lifespan when user policies change frequently. (This chapter uses Oracle Internet Directory identity plug-in as the example.)

   For example, on the Global Settings - Query Configuration page, under **Secure Search Configuration**, enter `0` for **Security Filter Lifespan (minutes)**.

   Valid values for the security filter lifespan are between 0 minutes (no cache) and 526500 minutes (cache for one year).
3. To index everything, you must force a full crawl for each source; that is, you must change the existing incremental crawl schedule for each source to first process ALL documents.

   This step is very important, in that searching does not work unless the content is first indexed completely.

   Go to the Home - Schedules page, select the source schedule, and click **Edit** to force a full crawl.

After each source has been crawled, go back to the same page and change the crawl policy back to incremental (index documents that have changed since the previous crawl). Also, in the Frequency section of the page, select a non-manual type for running incremental crawl (for example, weekly or daily).

Note: Before the first crawl of the Content Server, remember to go to the Content Server Administration page, select **SES Crawler Export**, and take a snapshot. For more information, see Section 18.6.1, "Setting Up Oracle WebCenter Content Server for Oracle SES."

18.6.4 Setting Up WebCenter Portal Framework Applications for Oracle SES

This section describes how to configure WebCenter Portal Framework applications to work with Oracle SES.

Make sure that you have created and configured the connection between your Portal Framework application and Oracle SES, specifying the Federation Trusted Entity, and optionally specifying a source group.

The Oracle SES crawlers are enabled by default in Portal Framework applications.

> **See Also:** "Setting Up Connections for Search" in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*

18.6.4.1 Configuring Portal Framework Applications After Deployment

After a Portal Framework application has been deployed to a managed server, you can configure it using WLST or Fusion Middleware Control. This section contains the following topics:

- Section 18.6.4.1.1, "Modifying Search Parameters Using WLST"
- Section 18.6.4.1.2, "Configuring Oracle SES Version Using WLST"
- Section 18.6.4.1.3, "Configuring Search Crawlers Using WLST"
- Section 18.6.4.1.4, "Configuring Search Parameters and Crawlers Using Fusion Middleware Control"

18.6.4.1.1 Modifying Search Parameters Using WLST

Use the WLST command `setSearchConfig` to modify search parameters post deployment.

Example 18–12 shows how to specify a data group (also known as source group) under which you search Oracle SES.

Example 18–12 Set a Source Group

```
setSearchSESConfig(appName='myPortalApp',
                   dataGroup='mySources')
```

where `dataGroup` is the source group you create in Section 18.6.3.5, "Additional Oracle SES Configuration."

Example 18–13 shows how to increase the number of search results displayed. Five is the default setting for the number of search results displayed in Oracle SES results, but result sets generally are larger than five.

Example 18–13 Increase Number of Search Results Displayed

```
setSearchConfig(appName='myPortalApp',
                numResultsMain=10)
```

Example 18–14 shows how to configure the maximum time that a tool/service is allowed to execute a search (in ms). When a service times out largely depends on the system load. If you consistently get time out errors, then you can adjust this parameter.

Example 18–14 Configure Maximum Time WebCenter Waits for Search Results

```
setSearchConfig(appName='myPortalApp',
                executionTimeout=10000)
```

For command syntax and examples, see the "setSearchConfig" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

18.6.4.1.2 Configuring Oracle SES Version Using WLST You must run the `setSESVersion` WLST command to obtain and store version information for the Oracle SES instance associated with your default connection. This command enables faceted search and the Tools and Services - Search administration page, which is necessary for customizing search settings with Oracle SES 11.2.2.2.

If the Portal Framework application is deployed to JDeveloper's integrated server, then you must run the `setSESVersion` WLST command after starting the application. Version data is not persisted across each instance of the integrated server. Also, you must run the `setSESVersion` command before accessing the site of the Portal Framework application. For example:

1. Start the integrated server and the Portal Framework application.
2. Run the `setSESVersion` command.
3. Access the site.

To confirm that the Oracle SES version is set correctly, run the `listSESVersion` WLST command.

Example 18–15 shows these commands. For full command syntax and examples, see the "setSESVersion" and "listSESVersion" sections in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

Example 18–15 Enable Facet Query and the Tools and Services Search Admin Page

```
setSESVersion(appName='myPortalApp',
 sesUrl='http://myhost.com:5720/search/api/admin/AdminService',
 sesSchema='searchsys', sesPassword='password'
 listSESVersion(appName='myPortalApp',
 sesUrl='http://myhost.com:5720/search/api/admin/AdminService')
```

18.6.4.1.3 Configuring Search Crawlers Using WLST You can use WLST commands to create crawlers and to start, stop and delete crawler schedules post deployment. These commands let you crawl new data in Oracle SES or delete old crawlers if the configuration data changes.

Example 18–16 and Example 18–17 show some of these commands. For more information, see the "Search - Oracle SES Search Crawlers" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

Example 18–16 Create a Documents Crawler in WLST

```
createDocumentsCrawler(
```

```
appName='myPortalApp', host='myWebcenterHost', port='myWebcenterPort',
sesUrl='http://mySEShost.com:7777/search/api/admin/AdminService',
sesPassword='mySESAdminPassword',
configUrl='http://myContentServerHost:myContentServerPort/cs/idcplg?IdcService=SES
_CRAWLER_DOWNLOAD_CONFIG&source=default',
user='ContentServer_crawl_user', password='ContentServerCrawlPassword',
scratchDir='/tmp',
httpEndpoint='http://myContentServerHost:myContentServerPort/cs/idcplg',
displayUrl='http://myContentServerHost:myContentServerPort/cs', realm='Idc
Security /cs/idcplg',
authUserIdFormat='authentication-id-format', pipelineName='Document-pipeline',
crawlingMode='ACCEPT_ALL', recrawlPolicy='PROCESS_CHANGED', freqType='MANUAL',
startHour=1, hoursBetweenLaunches=1, startDayOfWeek='MONDAY', startDayOfMonth=1,
daysBetweenLaunches=1, weeksBetweenLaunches=1, monthsBetweenLaunches=1,
sesSchema='eqsys')
```

Example 18–17 Create a Discussions Crawler in WLST

```
createDiscussionsCrawler(
appName='myPortalApp', host='myWebcenterHost', port='myWebcenterPort',
sesUrl='http://mySEShost.com:7777/search/api/admin/AdminService',
sesPassword='mySESAdminPassword',
dbConnString='jdbc:oracle:thin:@database-host:database-port:database-sid',
user='Jive-crawler-schema', password='Jive-crawler-schema-pw',
authUserIdFormat='authentication-id-format', crawlingMode='ACCEPT_ALL',
recrawlPolicy='PROCESS_ALL', freqType='MANUAL', startHour=1,
hoursBetweenLaunches=1, startDayOfWeek='MONDAY', startDayOfMonth=1,
daysBetweenLaunches=1, weeksBetweenLaunches=1, monthsBetweenLaunches=1,
sesSchema='eqsys')
```

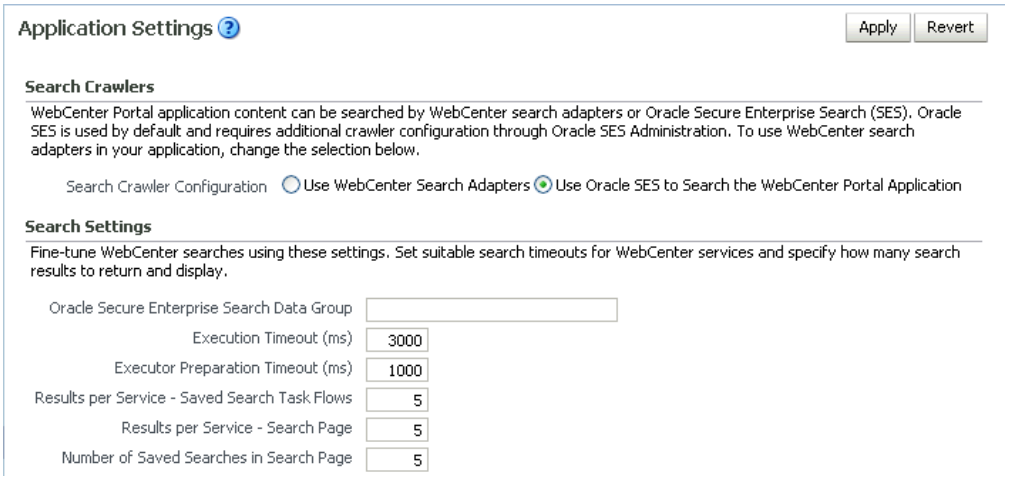
Notes:

- For *authentication-id-format*, use 'nickname' if the Identity Management plug-in on Oracle SES is set to Oracle Internet Directory; otherwise, use the value of the **Authentication Attribute** parameter on the Identity Management plug-in on Oracle SES.
- For *database-host*, use Oracle WebCenter Portal's Discussion Server database host name
- For *Jive-crawler-schema*, use the Discussions server crawler schema name. Determine the prefix from RCU, and use *rcu-prefix*_DISCUSSION_CRAWLER.
- For sesSchema, the default value is searchsys, which is the default admin user name for Oracle SES 11.2.2.2; however, a different name may have been specified during installation. If you are using Oracle SES 11.1.*, then set this parameter to eqsys.
- To effect WLST changes, you must restart the managed server on which the application is deployed (by default, WC_Spaces). For more information, see the "Starting and Stopping Managed Servers Using WLST" section in *Oracle Fusion Middleware Administrator's Guide*.

18.6.4.1.4 Configuring Search Parameters and Crawlers Using Fusion Middleware Control You can enable or disable Oracle SES and configure search settings using Fusion Middleware Control.

1. Log in to Fusion Middleware Control and navigate to the home page for your Portal Framework application. For more information, see Section 6.3, "Navigating to the Home Page for Portal Framework Applications."
2. From the **WebCenter Portal** menu, select **Settings** > **Application Configuration**.
3. In the **Search Crawlers** section, select to use Oracle SES, and click **Apply** (Figure 18–41).
4. Configure **Search Settings** as required, and click **Apply** (Figure 18–41).

Figure 18–41 Application Settings for Oracle Search



- **Oracle Secure Enterprise Search Data Group**: Specify the Oracle SES source group in which to search. If no value is provided, then everything in the Oracle SES instance is searched.
- **Execution Timeout**: Enter the maximum time that a tool/service is allowed to execute a search (in ms).
- **Executor Preparation Timeout**: Enter the maximum time that a tool/service is allowed to initialize a search (in ms).
- **Results per Service - Saved Search Task Flows**: Enter the number of search results displayed, per tool/service, in a Saved Search task flow.
- **Results per Service - Search Page**: Enter the number of search results displayed, per tool/service, for searches submitted from the main search page. Users can click Show All if they want to see all the results.
- **Number of Saved Searches in Search Page**: Enter the number of saved searches displayed in the Saved Search list (on the main search page).

You do *not* need to restart the managed server on which the Portal Framework application is deployed.

18.7 Troubleshooting Issues with Oracle SES

This section provides troubleshooting tips on administering Oracle SES. It includes the following topics:

- Section 18.7.1, "No Search Results Found"
- Section 18.7.2, "Search Failure Errors"

18.7.1 No Search Results Found

Problem

No search results are found.

Solution

Check the following:

- Oracle SES Connection
- Documents and Discussions Connections
- WebCenter Portal Crawl Configuration
- Oracle SES Configuration
- User Authentication
- Oracle SES Crawling
- Oracle SES Authorization

18.7.1.1 Oracle SES Connection

Confirm that you can access the Oracle SES SOAP URL and that connection properties to Oracle SES are correct.

For more information, see Section 18.4.1, "Testing the Connection to Oracle SES."

18.7.1.2 Documents and Discussions Connections

Confirm that connections exist in WebCenter Portal to the Content Server and the discussions server.

The Oracle SES log shows if a WebCenter Portal component is excluded from the search. Locate the search log file on the Oracle SES instance and check the log file for `totalSearchTime`.

When nothing is excluded (that is, Oracle SES is enabled for all WebCenter Portal components), the line looks similar to the following:

```
req=Search userName=vicki totalSearchTime=1150ms userQuery=0712>
```

When Oracle SES is not enabled for Documents, Discussions, and Announcements, the lines look similar to the following:

```
req=Search userName=vicki totalSearchTime=1133ms userQuery=0712
-wc_serviceId:oracle.webcenter.doclib
```

```
-wc_serviceId:oracle.webcenter.collab.forum
-wc_serviceId:oracle.webcenter.collab.announcement>
```

18.7.1.3 WebCenter Portal Crawl Configuration

Use Fusion Middleware Control or WLST to confirm that Oracle SES is enabled in WebCenter Portal, as described in Section 18.5.1, "Setting Up WebCenter Portal for Oracle SES."

18.7.1.4 Oracle SES Configuration

1. Confirm that you have installed all required patches for Oracle SES. For the latest information on required patches, see the "Back-End Requirements for Search" section in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal* and the Release Notes.
2. Confirm that Oracle SES is configured with an identity management system to validate and authenticate users. Also confirm that WebCenter Portal and Oracle SES use the same identity management system, such as Oracle Internet Directory. All repositories you are using (such as WebCenter Portal, WebCenter Content Server, and Oracle WebCenter Portal's Discussion Server) must share the same user base as Oracle SES.

   Additionally, each Oracle SES instance must have a trusted entity for allowing WebCenter Portal end users to be securely propagated at search time. For more information, see Section 18.3.2, "Oracle SES - Configuration."

   To test the Oracle SES is connection with a federated trusted entity user, see Section 18.4.1, "Testing the Connection to Oracle SES."

18.7.1.5 User Authentication

Confirm that the user exists (that is, confirm that the user can log on) in WebCenter Portal identity plug-ins, Oracle SES, and all configured data repositories, such as the Content Server and the discussions server.

An Oracle SES proxy login error in the WebCenter Portal diagnostic log looks similar to the following:

```
Received status "failed" during proxy login with application entity "weblogic" to
Oracle SES at [http://host:port/search/query/OracleSearch], as search user
"vicki". Defaulting to public.
```

18.7.1.6 Oracle SES Crawling

Confirm that Oracle SES crawled successfully in all sources.

1. In the Oracle SES administration tool, go to the Home - Schedules tab. Click the **Log File** icon to display the log file for the source. To obtain the location of the full log, click the **Status** link. The Crawler Progress Summary and Log Files by Source section displays the full path to the log file.
2. If Oracle SES fails to log in to the Content Server crawl endpoint due to an authentication error, then the following errors are logged:

   ```
   EQP-60303: Exiting saxthread due to errors

   EQP-80330: Unrecognized QName
   <http://schemas.xmlsoap.org/soap/envelope/>:Envelope
   oracle.search.sdk.crawler.PluginException
   ```

3. For the Oracle WebCenter source, verify if the rsscrawl servlet is unavailable. For example:

```
FATAL      main            EQP-80309: Exception while opening a stream to the
URI: https://example.com:port/rsscrawl?command=GetControl
```

4. For the Content Server source, verify if the password is invalid. For example:

```
-1 Admin credentials passed in were not valid - Rejecting request.
```

5. Monitor the crawl process in the Oracle SES administration tool with a combination of the following:
   a. Check the crawler progress and status on the Home - Schedules page. (Click Refresh Status.) From the Status page, you can view statistics of the crawl.
   b. Monitor your crawler statistics on the Home - Schedules - Crawler Progress Summary page and the Home - Statistics page.
   c. Monitor your search statistics on the Home - General page and the Home - Statistics page.

   See *Oracle Secure Enterprise Search Administrator's Guide* for tips to tune crawl performance.
6. Additionally, examine snapshots and datafeeds on the Content Server instance, and examine the Oracle WebCenter Portal's Discussions Server database.

18.7.1.7 Oracle SES Authorization

1. In the Oracle SES administration tool, go to the Home - Sources tab.
2. Click the **Edit** icon for each source to see source configuration tabs.
3. Click the **Authorization** tab to confirm the authorization connection string, user name, password, and authorization user ID format.
4. Examine the Oracle SES log file (described in a previous step). Look for phrases including a URL value. For example, the URIHandler initialized for the URI:

```
http://host:8888/sesUserAuth?userId=someone
```

For detailed information on the Oracle SES administration tool, see the Oracle SES documentation included with the product. (This is listed in the WebCenter Portal product area on the Oracle Fusion Middleware documentation library.)

See Also:

- Section 18.5.4.2, "Setting Up Oracle SES to Search Documents"
- Section 18.5.4.3, "Setting Up Oracle SES to Search Discussions and Announcements"
- Section 18.5.4.4, "Setting Up Oracle SES to Search Portals, Lists, People, and Page Metadata"

For Portal Framework applications:

- Section 18.6.3.2, "Setting Up Oracle SES to Search Documents"
- Section 18.6.3.3, "Setting Up Oracle SES to Search Discussions and Announcements"

18.7.2 Search Failure Errors

Problem

Search failure messages may appear inconsistently after a search. For example, when connecting to database `jdbc:oracle:thin:@host:1521/sid`, user: `PREFIX_DISCUSSIONS_CRAWLER`:

```
Search failure: time out error
Search failure: problem preparing search executor
Search failure: problem with execution
```

Solution

Wait a moment, and try the search again. These messages appear when the service times out, which largely depends on the system load. If the time out error persists, adjust the `executionTimeout` parameter in the `setSearchConfig` command.

For command syntax and examples, see the "setSearchConfig" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

18.7.3 Cannot Grant View Permissions to WebCenter Portal

Problem

You get the following error when granting `"view"` permissions, as described in Section 18.5.1, "Setting Up WebCenter Portal for Oracle SES."

```
Command FAILED, Reason: javax.naming.directory.AttributeInUseException: [LDAP: e
rror code 20 - uniquemember attribute has duplicate value.]; remaining name 'orc
lguid=F0CC506017B711DFBFFED9EA6A94EAEC,cn=Permissions,cn=JAAS Policy,cn=webcente
r,cn=wc_domain,cn=JPSContext,cn=jpsroot_webcenter_dadvmc0057'
```

Solution

This error appears if the permission is granted. Ignore the error.

18.7.4 Restricting Oracle SES Results by Source Group or Source Type

Problem

You want to restrict search results by source group or source type.

Solution

In the Oracle SES administration tool, navigate to the Home - Sources - Customize Federated Source - Search Restrictions page to set search restrictions.

Alternatively, use filters, where each filter is a restriction on search result.

For detailed information about using Oracle SES, see the Oracle SES documentation on the Oracle Fusion Middleware documentation library (in the WebCenter Portal product area).

18.7.5 Search Results Do Not Include Secured Resources

Problem

Search results do not include secured resources. One cause is that the proxy login of WebCenter Portal users failed in Oracle SES. An Oracle SES proxy login error in the WebCenter Portal diagnostic log looks similar to the following:

```
Received status "failed" during proxy login with application entity "weblogic" to
Oracle SES at http://host:port/search/query/OracleSearch, as search user "vicki".
Defaulting to public.
```

Solution

Confirm that Oracle SES is configured with an identity management system to validate and authenticate users.

Also confirm that WebCenter Portal and Oracle SES use the same identity management system, such as Oracle Internet Directory. All repositories (such as WebCenter Portal, WebCenter Content Server, and Oracle WebCenter Portal Discussions Server) must share the same user base as Oracle SES.

Additionally, each Oracle SES instance must have a trusted entity for allowing WebCenter Portal end users to be securely propagated at search time.

Problem

Search results do not include secured resources. Another cause is that authorization endpoints are not configured correctly. Locate the search log file on the Oracle SES instance. Look for phrases including the URL value. For example:

```
EQP-80309: Exception while opening a stream to the URI:
http://<host>:<port>/sesUserAuth?userId=<end-user-name>

QueryFilterPlugin returned null or empty array value for security attribute
"WCSECATTR". Values required for all security attributes.
```

Solution

1. In the Oracle SES administration tool, go to the Home - Sources tab.
2. Click the **Edit** icon for the source to see source configuration tabs.
3. Click the Authorization tab to confirm the authorization connection string, user name, password, and authorization user ID format.

Problem

Search results do not include secured resources. Yet another cause is that authorization endpoints are not returning authorization data.

Locate the search log file on the Oracle SES instance. Look for phrases including a URL value. For example, the URIHandler initialized for the URI:

```
http://host:8888/sesUserAuth?userId=someone
```

Solution

Reduce the number of crawl sources.

18.7.6 Search Results Do Not Include Documents

Problem

Search results do not include documents. Crawling of Content Server documents fails.

Solution

1. In the Oracle SES administration tool, go to the Home - Schedules tab.
2. Click the **Log File** icon to display the log file for the source. To obtain the location of the full log, click the **Status** link.

3. The Crawler Progress Summary and Log Files by Source section display the full path to the log file. If Oracle SES fails to log in to the Content Server crawl endpoint due to an authentication error, then the following errors are logged:

```
EQP-60303: Exiting saxthread due to errors,
EQP-80330: Unrecognized QName
<http://schemas.xmlsoap.org/soap/envelope/>:Envelope
oracle.search.sdk.crawler.PluginException
```

4. Update the configuration parameters of the Content Server crawl source.

18.7.7 Search Results Do Not Include Discussions and Announcements

Problem

In the crawl source, the **Single Record Query** parameter is set to true on the Authorization tab.

Solution

Set the **Single Record Query** parameter to false.

Problem

The identity management system uses mixed case user name, but Oracle WebCenter Portal's Discussions Server (Jive) database uses all lowercase user name. The authorization query for the crawl source must apply the `LOWER()` function to user name parameters.

Solution

Confirm that the authorization query looks like the following statement:

```
SELECT forumID as WCSECATTR FROM AUTHCRAWLER_FORUM_VW WHERE (username) = LOWER(?)
UNION SELECT DISTINCT -1 as WCSECATTR FROM AUTHCRAWLER_FORUM_VW
```

Note: It is not recommended to convert the actual user name column to lowercase in the query; for example, `WHERE LOWER(username) = LOWER(?)`. Adding a function to a possibly indexed column could affect performance.

18.7.8 Search Results Do Not Include Recently Added Resources

Problem

A new resource was created recently, but search results do not include the new resource.

Solution

New resources must be crawled and indexed before they can be returned in search results. Crawl schedules are run periodically to index new content. If new resources are created often, then increase the frequency of the crawl schedule. If new resources need to be crawled immediately, then start that crawl schedule manually.

18.7.9 Search Results Do Not Reflect Authorization Changes

Problem

Some resources are accessible to more users due to authorization changes in WebCenter Portal. For example, resources in a portal are now accessible to all authenticated users. The affected users cannot search for those resources.

Solution

Authorization data is cached in Oracle SES. The cache is invalidated according to the Security Filter Lifespan global setting in Oracle SES. The default value is 1 day or 1440 minutes. Adjust the value according to the general frequency of changes to authorization data.

18.7.10 Search Results Do Not Include Resources Available to Wide Audience

Problem

A portal is publicly accessible, but unauthenticated users cannot see portal resources in search results.

Solution

By default, view access of resources is granted to portal members only, even if the portal is accessible to the public. View access of resources must be granted to non-members explicitly. Go to the portal settings page, select the Role tab and the intended role, and check view access to resources.

19

Managing Subscriptions and Notifications

This chapter describes how to administer subscriptions and notifications. As an administrator, you can create and, potentially, enforce application-wide defaults for application-level subscriptions; specify a connection type that identifies the server that will handle notification delivery; and set and get Notifications messaging configuration details using WLST commands.

This chapter includes the following topics:

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console and the `Administrator` role in the deployed application:

- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.
- **Portal Framework application**: `Administrator` role granted through the Administration Console.

For more information about roles and permissions, see Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

19.1 About Subscriptions and Notifications

In WebCenter Portal, subscriptions and notifications provide users with a means of subscribing to the types of services and application objects in which they have a particular interest. Consequently, users receive timely notice of the changes that affect their subscribed services and objects from their selected messaging channels.

See Also: For information on adding notifications functionality to a portal, see the "Adding Notifications to a Portal" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Always use the Fusion Middleware Control or WLST command-line tool to review and configure back-end services for WebCenter Portal and Portal Framework applications. Any changes you make to WebCenter Portal or Portal Framework applications, post deployment, are stored in MDS metadata store as customizations. See Section 1.3.5, "Oracle WebCenter Portal Configuration Considerations."

Note: Most changes you make to WebCenter Portal tools and services configuration through Fusion Middleware Control or using WLST are not dynamic. For your changes to take effect, you must restart the managed server in which the application is deployed. See Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

19.2 Setting Up Default Subscription Preferences

WebCenter Portal users set their personal Subscriptions preferences through WebCenter Portal's Preferences dialog. Before this happens, the WebCenter Portal administrator can set default values that determine the application-level subscription options that are available to all users and whether those defaults can be changed.

This section provides an overview of Subscription defaults and steps you through the process of setting default values.

This section includes the following subsections:

- Section 19.2.1, "About Subscription Defaults"
- Section 19.2.2, "Setting Subscription Defaults"
- Section 19.2.3, "Setting Subscriptions Preferences in WebCenter Portal"

19.2.1 About Subscription Defaults

Administrator-level Subscription preferences are set in a custom XML file that you create and then use to supersede the file that is provided for this purpose out of the box (`notification-service-settings.xml`). The settings in the custom XML file are analogous to the application-level subscriptions settings available to users through Subscription Preferences in the WebCenter Portal (for more information, see the "Subscribing to the Application, to Portals, and to Objects" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.)

Each setting provides three attributes:

- `id`—for specifying the service ID:
  - `oracle.webcenter.peopleconnections.connections`, the Connections feature of the People Connections service
  - `oracle.webcenter.peopleconnections.wall`, the Message Board feature of the People Connections service
  - `oracle.webcenter.peopleconnections.kudos`, the Feedback feature of the People Connections service
  - `oracle.connections. community`, portal membership management
- `subscription-enabled`—for specifying the default value for the preference option: `true` or `false`

Tip: Rather than enabling or disabling the entire subscription capability, the `subscription-enabled` attribute merely sets the initial state of the preference option. For example, if `subscription-enabled="true"`, then the associated subscription option is selected by default in the WebCenter Portal's Preferences dialog. If `subscription-enabled="false"`, then the associated subscription option is deselected by default in the dialog.

- `end-user-configurable`—for enabling users to change the established default or preventing users from doing so: `true` or `false`

These attributes work together to determine the initial state of the **General Subscriptions** tab on the **Subscriptions** panel in the WebCenter Portal's Preferences dialog (Figure 19–1).

Figure 19–1 General Subscriptions Tab on the Subscriptions Page

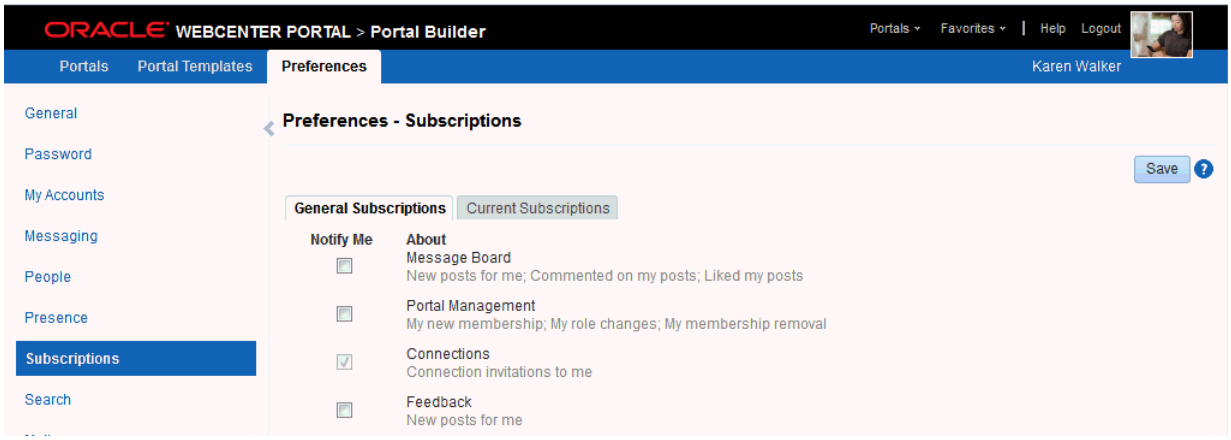


Table 19–1 illustrates the effect of custom administrator-level subscriptions settings on the appearance of the **General Subscriptions** tab.

Table 19–1 Effect of Administrator Defaults on Subscriptions Preferences

subscription-enabled[1]	end-user-configurable	Option in Preferences
True	True	Rendered normally, check box is selected
True	False	Grayed out, check box is selected
False	True	Rendered normally, check box is deselected
False	False	Hidden, check box is hidden

[1] Rather than enabling or disabling the entire subscription capability, the `subscription-enabled` attribute merely sets the initial state of the preference option. For example, if `subscription-enabled="true"`, then the associated subscription option is selected by default in WebCenter Portal's Preferences. If `subscription-enabled="false"`, then the associated subscription option is deselected by default.

Tip: In Table 19–1, the most typical scenario for most notifications is depicted in row three.

Table 19–2 lists the types of actions that can trigger an application-level notification and associates them with their related service ID.

Table 19–2 Application-Level Activities that Can Trigger Notifications

Activity	Related Service ID
A user sends you an invitation to connect	`oracle.webcenter.peopleconnections.connections`
Your portal role changes, for example, from *participant* to *moderator*	`oracle.webcenter.community`
You are added as a member of a portal	`oracle.webcenter.community`
Your portal membership is removed	`oracle.webcenter.community`
A user posts a message to your Message Board	`oracle.webcenter.peopleconnections.wall`
A user likes your post on another user's Message Board	`oracle.webcenter.peopleconnections.wall`
A user comments on your post on another user's Message Board	`oracle.webcenter.peopleconnections.wall`
A user posts feedback for you	`oracle.webcenter.peopleconnections.kudos`

19.2.2 Setting Subscription Defaults

To set defaults for application-level Subscription preferences:

1. Navigate to a directory with a path that contains `/oracle/webcenter/notification`, and create the folder `custom`.

   Tip: The directory structure can start or end with any directory or directories, as long as it has `/oracle/webcenter/notification/custom` in the path.

2. In the `custom` folder, or in any subdirectory under `/oracle/webcenter/notification/custom/`, create the file `notification-service-settings.xml`.
3. In the XML file, enter values for all application-level subscription options.

   Example 19–1 provides sample content for an application-wide subscription preferences setting file and an example of each required option.

Example 19–1 Sample Subscriptions Settings XML File

```
<notification-service_settings xmlns="http://xmlns.oracle.com/webcenter/notification">
  <subscription-settings>
    <service id="oracle.webcenter.peopleconnections.connections" subscription-enabled="true"
      end-user-configurable="false"/>
    <service id="oracle.webcenter.peopleconnections.wall" subscription-enabled="false"
      end-user-configurable="true"/>
    <service id="oracle.webcenter.peopleconnections.kudos" subscription-enabled="false"
      end-user-configurable="true"/>
    <service id="oracle.webcenter.community" subscription-enabled="true"
      end-user-configurable="true"/>
  </subscription-settings>
</notification-service_settings>
```

Note: If an option is not provided, the default values `false/false` are assigned for the service.

4. Run the WLST command `importMetadata()`, and import the directory content into your metadata store.

   See Also: For information about running WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands." For information about the `importMetadata()` command (and other WLST commands), see the "importMetadata" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

   For example:

```
wls: /wc_domain/serverConfig> importMetadata(application='webcenter',
  server='serverName', fromLocation='directoryPath', docs='/**')
```

   Where:

   - *`application`* is the name that identifies your WebCenter Portal deployment
   - *`serverName`* is the name of the server on which WebCenter Portal is running
   - *`directoryPath`* is the directory path under which `oracle/webcenter/notification/custom/<any_sub_dir_after_this>/notification-service-settings.xml` is located.

     For example, if the directory path to `notification-service-settings.xml` is `/scratch/mydir/oracle/webcenter/notification/custom`, enter `/scratch/mydir` for `directoryPath`.
   - *`docs`* identifies the content to be imported, in this example, the path and files that fall under `directoryPath`.

Table 19–3 describes the effect of various combinations of settings for the service ID `oracle.webcenter.peopleconnections.connections`.

Table 19–3 Effects of Subscription Configurations for Connections

subscription-enabled	end-user-configurable	Effect
true	true	■ The subscribing user receives a notification message when another user sends the user an invitation to connect. ■ The user can change this default.
true	false	■ The subscribing user receives a notification message when another user sends the user an invitation to connect. ■ The user cannot change this default.[1]
false	true	■ The subscribing user does not receive a notification message when another user sends the user an invitation to connect. ■ The user can change this default.
false	false	■ The subscribing user does not receive a notification message when another user sends the user an invitation to connect. ■ The option for changing this default is hidden.

[1] This is the out-of-the-box default

Table 19–4 describes the effect of various combinations of settings for the service ID `oracle.webcenter.peopleconnections.wall`.

Table 19–4 Effects of Subscription Configurations for Message Board

subscription-enabled	end-user-configurable	Effect
true	true	■ The subscribing user receives a notification message when another user posts a message on the user's Message Board, likes the user's Message Board post, or comments on the user's Message Board post. ■ The user can change this default.
true	false	■ The subscribing user receives a notification message when another user posts a message on the user's Message Board, likes the user's Message Board post, or comments on the user's Message Board post. ■ The user cannot change this default.
false	true	■ The subscribing user does not receive a notification message when another user posts a message on the user's Message Board, likes the user's Message Board post, or comments on the user's Message Board post. ■ The user can change this default.
false	false	■ The subscribing user does not receive a notification message when another user posts a message on the user's Message Board, likes the user's Message Board post, or comments on the user's Message Board post. ■ The option for changing this default is hidden.

Table 19–5 describes the effect of various combinations of settings for the service ID `oracle.webcenter.peopleconnections.kudos`.

Table 19–5 Effect of Subscription Configurations for Feedback

subscription-enabled	end-user-configurable	Effect
true	true	■ The subscribing user receives a notification message when another user leaves feedback for the user. ■ The user can change this default.
true	false	■ The subscribing user receives a notification message when another user leaves feedback for the user. ■ The user cannot change this default.
false	true	■ The subscribing user does not receive a notification message when another user leaves feedback for the user. ■ The user can change this default.
false	false	■ The subscribing user does not receive a notification message when another user leaves feedback for the user. ■ The option for changing this default is hidden.

Table 19–6 describes the effect of various combinations of settings for the service ID `oracle.webcenter.community`.

Table 19–6 Effect of Subscription Configurations for Portal Management

subscription-enabled	end-user-configurable	Effect
true	true	■ The subscribing user receives a notification message when the user's portal membership role changes, the user is added as a member of a portal, or the user is removed as a member of a portal. ■ The user can change this default.
true	false	■ The subscribing user receives a notification message when the user's portal membership role changes, the user is added as a member of a portal, or the user is removed as a member of a portal. ■ The user cannot change this default.
false	true	■ The subscribing user does not receive a notification message when the user's portal membership role changes, the user is added as a member of a portal, or the user is removed as a member of a portal. ■ The user can change this default.
false	false	■ The subscribing user does not receive a notification message when the user's portal membership role changes, the user is added as a member of a portal, or the user is removed as a member of a portal. ■ The option for changing this default is hidden.

19.2.3 Setting Subscriptions Preferences in WebCenter Portal

Individual users set their own subscription preferences in the WebCenter Portal's Preferences. Two Preferences pages are provided for this purpose:

- **Subscriptions**, where users subscribe to be notified about actions occurring with their portal memberships and the People Connections service (Connections, Message Board, and Feedback) and view and remove their application- and object-level subscriptions

  For more information, see the "Subscribing to the Application, to Portals, and to Objects" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

- **Messaging**, where users access controls for configuring their preferred messaging channels and filters (BPEL connection types only)

  For more information, see the "Establishing and Managing Your Messaging Channels and Filters" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

19.3 Setting Up Notifications

This section provides an overview of messaging connection types, describes prerequisites that must be in place before you can define a notification channel, and steps you through the process of setting up a notification channel for Notifications. It includes the following subsections:

- Section 19.3.5, "Specifying the Notifications Channel Using WLST"
- Section 19.3.6, "Example - Setting Up Mail Notifications for WebCenter Portal Using WLST"

19.3.1 About Connection Channels

The Notifications connection type determines the messaging channels that are available to users when they configure their own messaging preferences for Notifications in WebCenter Portal.

Use one of two possible connection types:

- **BPEL Server** provides three messaging channel options to users: mail, texting (SMS), and worklist
- **Mail Server** delivers notification messages exclusively through a mail server that is configured for WebCenter Portal

BPEL Server

Selection of a BPEL server presupposes that you have established a connection with a BPEL server in which the User Messaging Service (UMS) is available. For information about connecting to a BPEL server, see Chapter 20, "Managing Worklists."

When WebCenter Portal has `setSpacesWorkFlowConnectionName` set up, the **Manage Configuration** button becomes available on the **Messaging** panel in WebCenter Portal's Preferences.

> **Tip:** You should use the same connection you use for `setSpacesWorkFlowConnectionName` is used for Notifications, provided you use the BPEL Server for notifications.

Mail Server

Selection of a mail server presupposes that you have established a connection with a mail server. Additionally, the external application associated with the mail server connection must contain shared credentials. For information about connecting to a mail server, see Chapter 15, "Managing Mail."

When **Mail Server** is the selected connection type, the **Manage Configuration** button on the **Messaging** panel in WebCenter Portal's Preferences might or might not be grayed-out. This depends on whether you have set up `spacesWorkFlowConnection`. Regardless, when Mail Server is the selected connection type and you click the **Manage Configuration** button for Messaging preferences opens User Messaging Preferences, any changes you make are ignored.

> **See Also:** The "Establishing and Managing Your Messaging Channels and Filters" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*

19.3.2 Notification Prerequisites

Before you can define a connection type for Notifications, you must take the steps and consider the information provided in the following subsections:

- Section 19.3.2.1, "Installation"
- Section 19.3.2.2, "Configuration"
- Section 19.3.2.3, "Security"

- Section 19.3.2.4, "Limitations"

19.3.2.1 Installation

Installation requirements associated with Notifications change according to the type of connection you plan to select for Notifications messaging.

If you plan to use the User Messaging Service (UMS) through your BPEL connection for Notifications messaging, you should know that only the mail driver is installed by default. To make use of SMS and Worklist messaging channels, you must install drivers for these as well. For information about installing SMS and Worklist drivers for UMS, see the "Configuring Oracle User Messaging Service" chapter in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite*.

If you plan to use the Mail service for Notifications messaging, no Notifications-specific installation is required, but the Mail service must be configured as described in Chapter 15, "Managing Mail."

19.3.2.2 Configuration

Configuration prerequisites for Notifications also depend on the connection type you plan to select for Notifications messaging.

BPEL Server

If you plan for users to have messaging channel options—mail, texting (SMS), and Worklist—a connection to a BPEL server must be in place. Notifications uses the SOA installation for supporting multichannel notifications through the User Messaging Service (UMS). UMS is installed as a part of the SOA domain. Out of the box, only the email driver is configured. The SMS driver is available, but must be deployed. For the Worklist channel, the SOA domain must be extended through the Worklist driver extension template.

For more information see Chapter 20, "Managing Worklists," and the "Configuring Oracle User Messaging Service" chapter in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite*.

Mail Server

If you plan for users to always and only be notified through their mail, a connection to a mail server must be in place. Additionally, the external application associated with the mail server connection must contain shared credentials. For more information, see Chapter 15, "Managing Mail."

Mail notifications are sent in the preferred language specified for each user's profile. If the preferred language is not specified for a user, the server locale setting is used for mail notifications. For example, if the server is running on the Korean locale and the preferred language is not set for a user, the notification mail is in Korean.

For information about setting the preferred language, see the "Choosing Your Preferred Display Language" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

19.3.2.3 Security

There are no security considerations specifically associated with Notifications.

19.3.2.4 Limitations

Some activities create Notification tasks to be sent in the future. For example, if a user creates an announcement with an active date in the future, a notification task is created on the WebCenter Portal application server, so that a notification will be sent when the announcement becomes active. However, if the Mail service is used for Notifications, future Notification tasks are deleted if the WebCenter Portal application server is restarted.

UMS supports multiple messaging channels, including voice and instant messaging, that are not supported by Notifications. From UMS, Notifications consumes only mail, SMS, and Worklist.

19.3.3 Configuration Roadmap for Notifications

Figure 19–2 and Table 19–7 provide an overview of the prerequisites and tasks required to get the Notifications service working in WebCenter Portal applications.

Figure 19–2 Configuring the Notifications Service

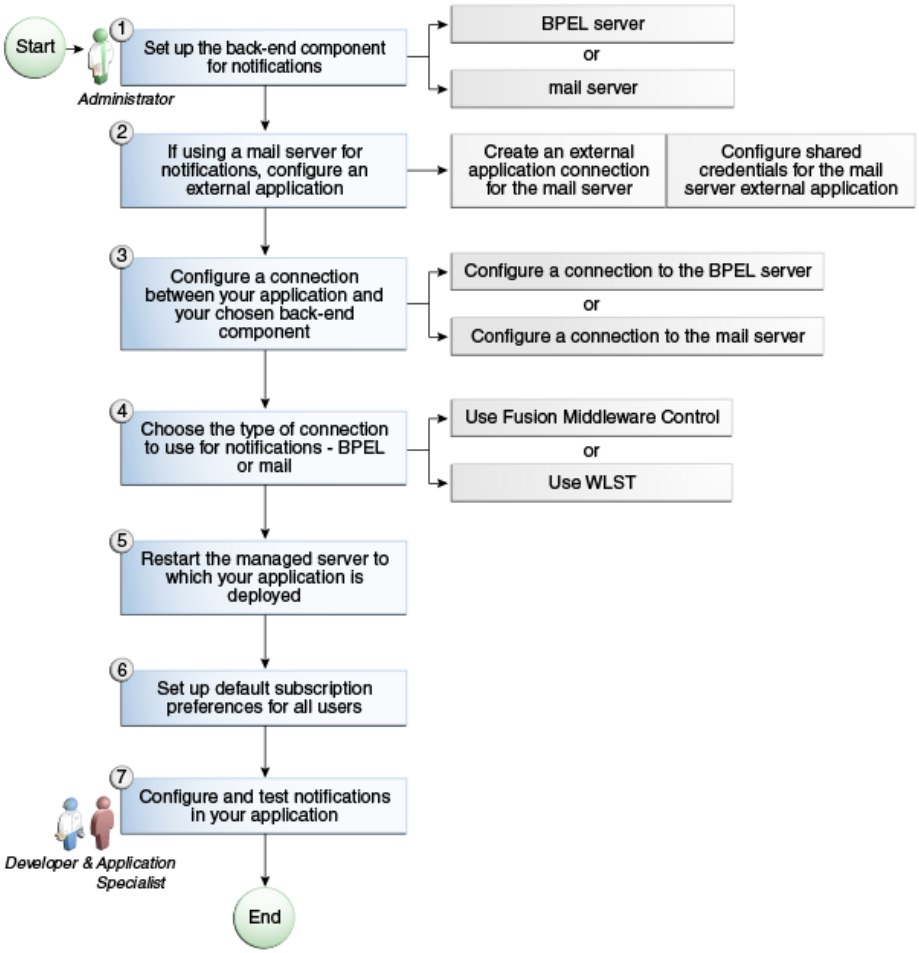


Table 19–7 Configuring Notifications

Actor	Task	Sub-task	Notes
Administrator	**1.** Set up the back-end component for Notifications. ■ Set up the BPEL server ■ Set up the mail server		
Administrator	**2.** (For mail server only) Configure an external application.	■ Create an external application connection for the mail server ■ Configure shared credentials for the mail server external application	
Administrator	**3.** Create or modify a connection between your WebCenter Portal application and your chosen back-end component: ■ Create a connection to the BPEL server ■ Create a connection to the mail server		
Administrator	**4.** Choose the type of connection to use for Notifications, either BPEL or Mail, by using one of the following tools: ■ Fusion Middleware Control ■ WLST		
Administrator	**5.** Restart the managed server to which your application is deployed.		For WebCenter Portal, restart `WC_Spaces`. For a Portal Framework application, restart the custom managed server where the application is deployed.
Administrator	**6.** Set up default subscription preferences for all users.		
Application Specialist/End User (WebCenter Portal) or Developer (Framework applications)	**7.** Configure and test Notifications in your applications: ■ WebCenter Portal (application specialist)/(end user) ■ Framework applications		

19.3.4 Specifying the Notifications Channel Using Fusion Middleware Control

To specify a Notifications message connection type with Fusion Middleware Control:

1. Log in to Oracle Fusion Middleware Control and navigate to the home page for WebCenter Portal.

   For more information, see Section 6.2, "Navigating to the Home Page for WebCenter Portal."

2. From the **WebCenter Portal** menu, select **Settings** > **Application Configuration**.

3. On the **Application Configuration** page, scroll down to **Notifications** (at the bottom of the page), and select a connection type to use for outbound notifications: either **BPEL Server** or **Mail Server**.
4. The next step depends on the selected connection type:

   If you select **BPEL Server**:

   a. From the **Connection Name** list, select the name you provided for the BPEL server when you set up that connection.

   b. In the **Sender Mail Address** field, enter a mail address from which all Notifications messages are sent. The sender mail address must match at least one driver that is configured to send messages from a corresponding domain.

   c. In the **Sender SMS Address** field, enter the four- to six-digit number that is used by the User Messaging Server (UMS) as the driver from which all Notifications messages are sent. The sender SMS address must match at least one driver that is configured to send messages from a corresponding domain.

   If you select **Mail Server**, select a mail connection from the **Connection Name** list.
5. Save your changes.
6. Restart the managed server on which WebCenter Portal is deployed to make your configuration changes take effect.

19.3.5 Specifying the Notifications Channel Using WLST

Use the WLST command `setNotificationsConfig` to configure the connection type used for notifications. For command syntax and examples, see the "setNotificationsConfig" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*. See also, the "getNotificationsConfig" section in the same guide.

> **See Also:** For information about how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Note: Updates to this configuration are stored in the MDS repository. For configuration changes to take effect, you must restart the managed server on which the application is deployed. For more information, see the "Starting and Stopping Managed Servers Using WLST" section in the *Oracle Fusion Middleware Administrator's Guide*.

19.3.6 Example - Setting Up Mail Notifications for WebCenter Portal Using WLST

This section provides an example of using WLST to set up Mail Notifications for WebCenter Portal using WLST commands.

First, the example shows you how to create an external application that is configured with shared credentials, and create a mail server connection that uses the external application. Next, the example shows you how to configure WebCenter Portal to send notifications on that mail connection, and finally how to set subscription options through user preferences.

1. At the WLST command prompt, connect to the Administration Server for WebCenter Portal.

```
connect('admin_user','mypassword','<servername>:7001')
```

2. Create an external application connection:

```
createExtAppConnection(appName='webcenter', name='NotificationSharedApp',
displayName= 'NotificationSharedApp')
```

   This command creates the connection named `NotificationSharedApp`.

   For more information, see the "createExtAppConnection" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

3. Configure shared credentials for the external application, `NotificationSharedApp`:

```
addExtAppCredential(appName='webcenter', name='NotificationSharedApp',
type='SHARED', username='john.doe@example.com', password='sharedpassword')
```

   Where `username` refers to the email account from which email notifications will be sent. This must be in the format `<user>@<domain of the mail server>`.

   Optionally, you may add the following fields that will be used while sending out the mail notification.

```
addExtAppField(appName='webcenter',name='NotificationSharedApp',fieldName='Emai
l Address',fieldValue='sender's_email_address',displayToUser=false)
addExtAppField(appName='webcenter',name='NotificationSharedApp',fieldName='Your
Name',fieldValue='sender's_display_name',displayToUser=false)
```

   For more information, see the "addExtAppCredential" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

4. Create a Mail connection:

```
createMailConnection(appName='webcenter',name='NotificationSharedConn',
         imapHost='<mailserver>',imapPort=143,
         smtpHost='<mailserver>',smtpPort=25,
         imapSecured=false,smtpSecured=false,
         appId='NotificationSharedApp',default=1)
```

   This creates a mail connection named `NotificationSharedConn`.

   For more information, see the "createMailConnection" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

5. Set Mail as the notifications channel:

```
   setNotificationsConfig(appName='webcenter', type='MAIL',
                          name='NotificationSharedConn')
```

   This sets `NotificationSharedConn` as the mail connection to use for sending notifications.

   For more information, see the "setNotificationsConfig" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

6. For the changes to take effect, restart the managed server on which WebCenter Portal is deployed (`WC_Spaces` by default).

7. Log in to WebCenter Portal, navigate to the **About** tab of the **Profile** page, and verify that your e-mail address is set in the **Email** field. This is to ensure that notifications are sent to the required e-mail address.

   If the e-mail address is not set, click **Edit**, then in the **Email** field, specify your e-mail address, and click **Save**.

8. Subscribe to the activities for which to receive notifications. For example, navigate to the Preferences page, click **Subscriptions**, and then select **Portal Management** to get notified about any membership or role changes.
9. Test your configuration by performing a subscribed activity. For example, change your role from Moderator to Participant to trigger a notification.

19.4 Creating and Applying Custom Notification Templates

The notification messages that users receive through Worklist or Mail have a default format for content and content presentation. As the application administrator, you can instead create and apply custom templates to provide your own formats for notification messages.

This section provides information about creating a custom template for notifications messages. It includes the following subsections:

- Section 19.4.1, "About Overwriting Default Notification Templates"
- Section 19.4.2, "Overwriting a Default Notifications Template"

19.4.1 About Overwriting Default Notification Templates

You can go through MDS using WLST commands to customize the layout and content of subscription-based notification messages by overwriting the files `defaultTemplate.xml` and `defaultTemplate_rtl.xml`—when right-to-left language support is required.

> **See Also:** For information about running WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

You can create your own version of these `xml` files, editing the CSS styles for tables (`label`, `value`, `background`) and footers (`note`). You can move such tags as `<payload>` and `<group-space-footer>` to change the layout. To modify the content of these tags, you can edit the `CDATA` section within `<html-format>`.

Note that the tag `<text-format/>` should always be present and empty. You can use the tag `<custom>` to add additional content, where the enclosed `<html-format>` with `CDATA` contains the new HTML content and `<text-format/>` remains empty.

Example 19–2 and Example 19–3 illustrate the default content of notification message template files. You can use these to formulate your custom files.

> **Note:** The default content of these files is very similar. The differences appear under the `<style>` tag, where alignment—either right or left—is specified.

Example 19–2 Default File defaultTemplate.xml

```
<?xml version="1.0"?>
<notification-template xmlns="http://xmlns.oracle.com/webcenter/notification">
     <!-- The CSS Style of the Notification -->
          <style>
               <text-format/>
               <html-format>
                    <![CDATA[
                         <style type="text/css">
```

```
                                .title  {font-size:1.2em; font-weight:bold;
                                     white-space:nowrap;}
                                .label  {text-align:right; margin-left:30px;
                                     padding-right:10px; white-space:nowrap;}
                                .value  {text-align:left; margin-right:20px;
                                     padding-left:10px; white-space:nowrap;
                                     width:100%;}
                                .note   {font-size:0.8em; color:#999999}
                                .background {background-color:#fcfcfc}
                           </style>
                      ]]>
                 </html-format>
            </style>


       <!-- The Subject line of the Notification -->
       <subject>
            <message-key>NOTIFICATION_SUBJECT</message-key>
       </subject>
       <group-space-subject>
            <message-key>GROUP_SPACE_SUBJECT_SUFFIX</message-key>
       </group-space-subject>
       <!-- Actual srvc-specific data. Provided/Overridden by srvc template -->
       <payload>
            <text-format/>
            <html-format/>
       </payload>


       <!-- Any generic/common footer to appear after service-specific payload -->
       <!-- Group Space footer - if applicable -->
       <group-space-footer>
            <text-format/>
            <html-format>
                 <![CDATA[
                      <p>
                           <a href="<token>groupSpaceUrl</token>" target="_blank">
                                <message-key>GO_TO_SPACE</message-key> <token>
                                     groupSpaceName</token>
                           </a>
                      </p>
                 ]]>
            </html-format>
       </group-space-footer>


       <!-- Unsubscribe footers -->
       <unsubscribe-footer>
            <text-format/>
            <html-format>
                 <![CDATA[
                      <hr/>
                      <p class="note">
                           <token>unsubscribeMessage</token>
                      </p>
                 ]]>
            </html-format>
       </unsubscribe-footer>
  </notification-template>
```

Example 19–3 Default File defaultTemplate_rtl.xml

```
<?xml version="1.0"?>
<notification-template xmlns="http://xmlns.oracle.com/webcenter/notification">
     <!-- The CSS Style of the Notification -->
          <style>
               <text-format/>
               <html-format>
                    <![CDATA[
                         <style type="text/css">
                              .title  {font-size:1.2em; font-weight:bold;
                                   white-space:nowrap;}
                              .label  {text-align:left; margin-right:30px;
                                   padding-left:10px; white-space:nowrap;}
                              .value  {text-align:right; margin-left:20px;
                                   padding-right:10px; white-space:nowrap;
                                   width:100%;}
                              .note   {font-size:0.8em; color:#999999}
                              .background {background-color:#fcfcfc}
                         </style>
                    ]]>
               </html-format>
          </style>


     <!-- The Subject line of the Notification -->
     <subject>
          <message-key>NOTIFICATION_SUBJECT</message-key>
     </subject>
     <group-space-subject>
          <message-key>GROUP_SPACE_SUBJECT_SUFFIX</message-key>
     </group-space-subject>
     <!-- Actual srvc-specific data. Provided/Overridden by srvc template -->
     <payload>
          <text-format/>
          <html-format/>
     </payload>


     <!-- Any generic/common footer to appear after service-specific payload -->
     <!-- Group Space footer - if applicable -->
     <group-space-footer>
          <text-format/>
          <html-format>
               <![CDATA[
                    <p>
                         <a href="<token>groupSpaceUrl</token>" target="_blank">
                              <message-key>GO_TO_SPACE</message-key> <token>
                                   groupSpaceName</token>
                         </a>
                    </p>
               ]]>
          </html-format>
     </group-space-footer>


     <!-- Unsubscribe footers -->
     <unsubscribe-footer>
          <text-format/>
          <html-format>
               <![CDATA[
```

```
                    <hr/>
                    <p class="note">
                         <token>unsubscribeMessage</token>
                    </p>
               ]]>
          </html-format>
     </unsubscribe-footer>
</notification-template>
```

19.4.2 Overwriting a Default Notifications Template

To overwrite an existing xml file to customize notification message formats:

1. Create a custom XML file with the name `defaultTemplate.xml` (or `defaultTemplate_rtl.xml`, for right-to-left language template).
2. Populate the custom file with your revised version of one of these default files.

   See Also: Example 19–2 and Example 19–3 show default file content.

3. Overwrite the original file, placing the custom file where the absolute path to the file contains the namespace `oracle/webcenter/notification/custom`.

   For example:

   ```
   /tmp/repository/oracle/webcenter/notification/custom/template/defaultTemplate.xml
   ```

4. Upload the custom file into WebCenter Portal's MDS repository by running the `importMetadata()` WLST command.

   For example:

   ```
   importMetadata(application='webcenter', server='WC_Spaces',
        fromLocation='template-file-location',
        docs='/oracle/webcenter/notification/custom/template/defaultTemplate.xml')
   ```

   The *`template-file-location`* points to the directory under which the fully qualified custom file is located. The fully qualified custom file is typically placed under the directory structure equivalent to its namespace. For example, consider a file that is created under the following namespace:

   ```
   /tmp/repository/oracle/webcenter/notification/custom/template/defaultTemplate.xml
   ```

   In such a case, the `fromLocation` is `/tmp/repository` because the remaining sub-directory consists of the namespace for the XML file. The namespace must have at least the path `/oracle/webcenter/notification/custom`.

   See Also: For information about the `importMetadata()` command (and other WLST commands), see the "importMetadata" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

5. Restart WebCenter Portal.

19.5 Testing the Notifications Connection

In general, Notifications depends on the underlying Mail or BPEL connection to be valid when the administrator sets it. If these connections prove to be valid, then, by extension, the Notifications connections requirements are met.

> **Tip:** For information about testing Mail connections, see Section 15.9, "Testing Mail Server Connections."

19.6 Troubleshooting Issues with Notifications

Problem

No notifications are received.

Solution

- If the log indicates that the Notification Sender is not configured, then it means the service is unable to find the connection to use.
- Ensure that Notifications is configured to use either a valid BPEL or MAIL connection. This can be verified through the `getNotificationsConfig()` WLST command (see Section 19.3.5, "Specifying the Notifications Channel Using WLST") or through the Fusion Middleware Control user interface (see Section 19.3.4, "Specifying the Notifications Channel Using Fusion Middleware Control").

Problem

Notifications is configured (BPEL or MAIL) correctly, but still no notifications.

Solution

Notifications relies on a valid BPEL or MAIL connection. Run the respective connection validations and troubleshooting scenarios as described in Chapter 15, "Managing Mail" or Chapter 20, "Managing Worklists."

Problem

MAIL or BPEL connections are set up appropriately, but still do not receive notifications.

Solution

Notifications are generated based on user subscriptions. Apart from notification for invitations to connect, which is configured out of the box, other notifications are generated only when a user has specifically subscribed. Ensure that the user has created subscriptions through his or her personal Preferences or through application- or object-level subscriptions. For more information, refer to the "Subscribing to the Application, to Portals, and to Objects" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

Problem

Users have set up their subscriptions, but still receive no notifications.

Solution

- Depending on how it is configured, Notifications delegates the delivery of notifications to BPEL/UMS or the Mail service. For the Mail service, ensure that the user's email address is configured. For UMS, look in Fusion Middleware

Control under the **Message Status** section of **User Messaging Service**, where you see the status of each outgoing message from UMS. For more information, see the "Monitoring Oracle User Messaging Service" chapter in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite*.

- For UMS, this problem could also mean that the configuration of the sender on the WebCenter Portal side does not match or find a corresponding driver on the UMS side. Ensure that the sender address (domain) allows UMS to match at least one driver for outbound messages.
- For the Mail service, ensure that the mail connection points to a shared connection as described in Section 19.3.1, "About Connection Channels."

Problem

For UMS configurations, users receive notifications on some channels but not on others.

Solution

This is most likely due to the way the user's messaging channels and filters are configured. For more information, see the "Establishing and Managing Your Messaging Channels and Filters" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

Problem

For UMS configurations, only mail-channel notifications are delivered, the Worklist channel does not work.

Solution

Ensure that the SOA domain is extended with the Worklist driver template as described in the "Configuring Oracle User Messaging Service" chapter in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite*.

20

Managing Worklists

This chapter describes how to configure and manage worklists for WebCenter Portal and Portal Framework applications.

Always use Fusion Middleware Control or WLST command-line tool to review and configure back-end servers for WebCenter Portal and Portal Framework applications. Any changes that you make to WebCenter Portal and Portal Framework applications, post deployment, are stored in MDS metatdata store as customizations. See Section 1.3.5, "Oracle WebCenter Portal Configuration Considerations."

> **Note:** Changes that you make worklist configuration, through Fusion Middleware Control or using WLST, are not dynamic, so you must restart the managed server on which WebCenter Portal is deployed for your changes to take effect. See Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

This chapter includes the following topics:

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console and the `Administrator` role in the deployed application:

- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.
- **Portal Framework application**: `Administrator` role granted through the Administration Console.

For more information about roles and permissions, see Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

20.1 Configuration Roadmaps for Worklist

Use the roadmaps in this section as a system administrator's guide through the configuration process:

- Section 20.1.1, "Roadmap - Configuring Worklist for WebCenter Portal"
- Section 20.1.2, "Roadmap - Configuring Worklist for Portal Framework Applications"

20.1.1 Roadmap - Configuring Worklist for WebCenter Portal

Figure 20–1 and Table 20–1 in this section provide an overview of the prerequisites and tasks required to use worklist in WebCenter Portal.

Figure 20–1 Configuring Worklist for WebCenter Portal

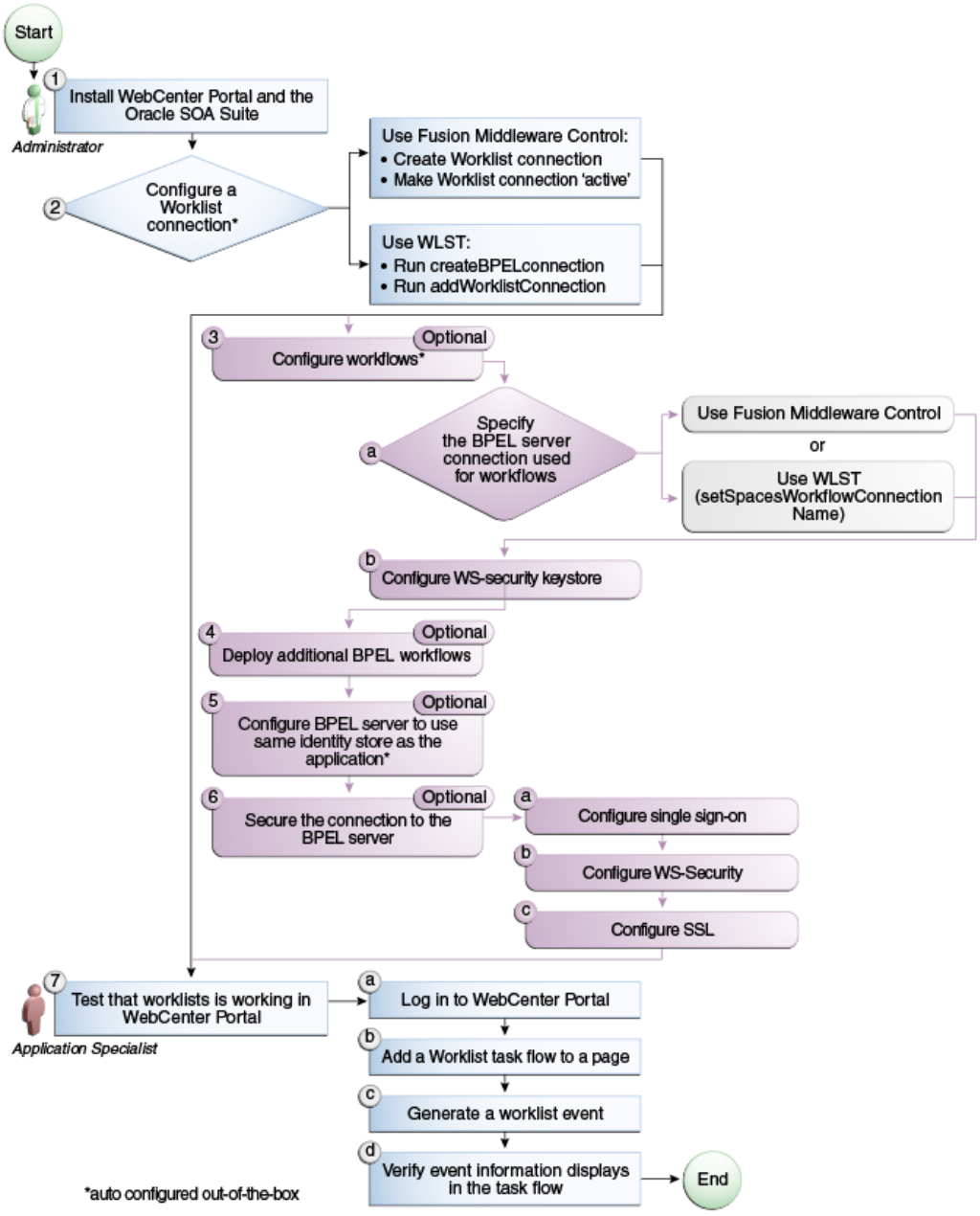


Table 20–1 Configuring Worklist for WebCenter Portal

Actor	Task	Sub-Task
Administrator	**1.** Install WebCenter Portal and the Oracle SOA Suite	To ensure the connection between WebCenter Portal and the BPEL server, make sure to do the following steps: **1.** Install the Oracle SOA Suite. **2.** Extend the SOA server domain with the `oracle.wc_composite_template_11.1.1.jar` template. For information, see the "Back-End Requirements For WebCenter Portal Workflows" section in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*.
Administrator	**2.** Configure a worklist connection using one of the following tools:[1] ■ Fusion Middleware Control (see Registering Worklist Connections Using Fusion Middleware Control) ■ WLST (see Registering Worklist Connections Using WLST)	When using Fusion Middleware Control: ■ **2.a** Create Worklist connection ■ **2.b** Make Worklist connection 'active' When using WLST: ■ **2.a** Run createBPELconnection ■ **2.b** Run addWorklistConnection

Table 20–1 (Cont.) Configuring Worklist for WebCenter Portal

Actor	Task	Sub-Task
Administrator	**3.** Configure WebCenter Portal workflows[1]	**3.a** Specify the BPEL server connection used for WebCenter Portal workflows using either of the following tools: ■ Fusion Middleware Control ■ WLST (`setSpacesWorkflowConnectionName`) **3.b** Configure WS-security keystore. ■ If you are not using the WebCenter Portal workflows to send group invitations to join a portal, then the security configuration is optional for worklists in an unsecured SAML environment. ■ Group membership invitations require the SOA server to invoke a secured WebCenter Web service on the WebCenter Server when a user accepts the group invitation to join a portal. For more information, see the "Back-End Requirements For WebCenter Portal Workflows" section in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*.
Administrator	**4.** (Optional) Deploy additional BPEL workflows	
Administrator	**5.** (Optional) Configure BPEL server to use same identity store as WebCenter Portal[1]	
Administrator	**6.** (Optional) Secure the connection to the BPEL server	**6.a** Configure single sign-on **6.b** Configure WS-Security **6.c** Configure SSL

Table 20–1 (Cont.) Configuring Worklist for WebCenter Portal

Actor	Task	Sub-Task
End User	**7.** Test that worklist is working in WebCenter Portal	**7.a** Log in to WebCenter Portal **7.b** Add a Worklist task flow to a page (see the "Adding a Worklist Task Flow to a Page" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*. **7.c** Generate a worklist event (see the "About Worklists" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*. **7.d** Verify event information displays in the task flow

[1] Auto configured out-of-the-box

See Also: Section 20.6, "Configuring WebCenter Portal Workflow Notifications to be Sent by Email" for steps to configure email notifications for your workflows.

20.1.2 Roadmap - Configuring Worklist for Portal Framework Applications

Figure 20–2 and Table 20–2 in this section provide an overview of the prerequisites and tasks required to use worklist in Portal Framework applications.

Figure 20–2 Configuring Worklist for Portal Framework Applications

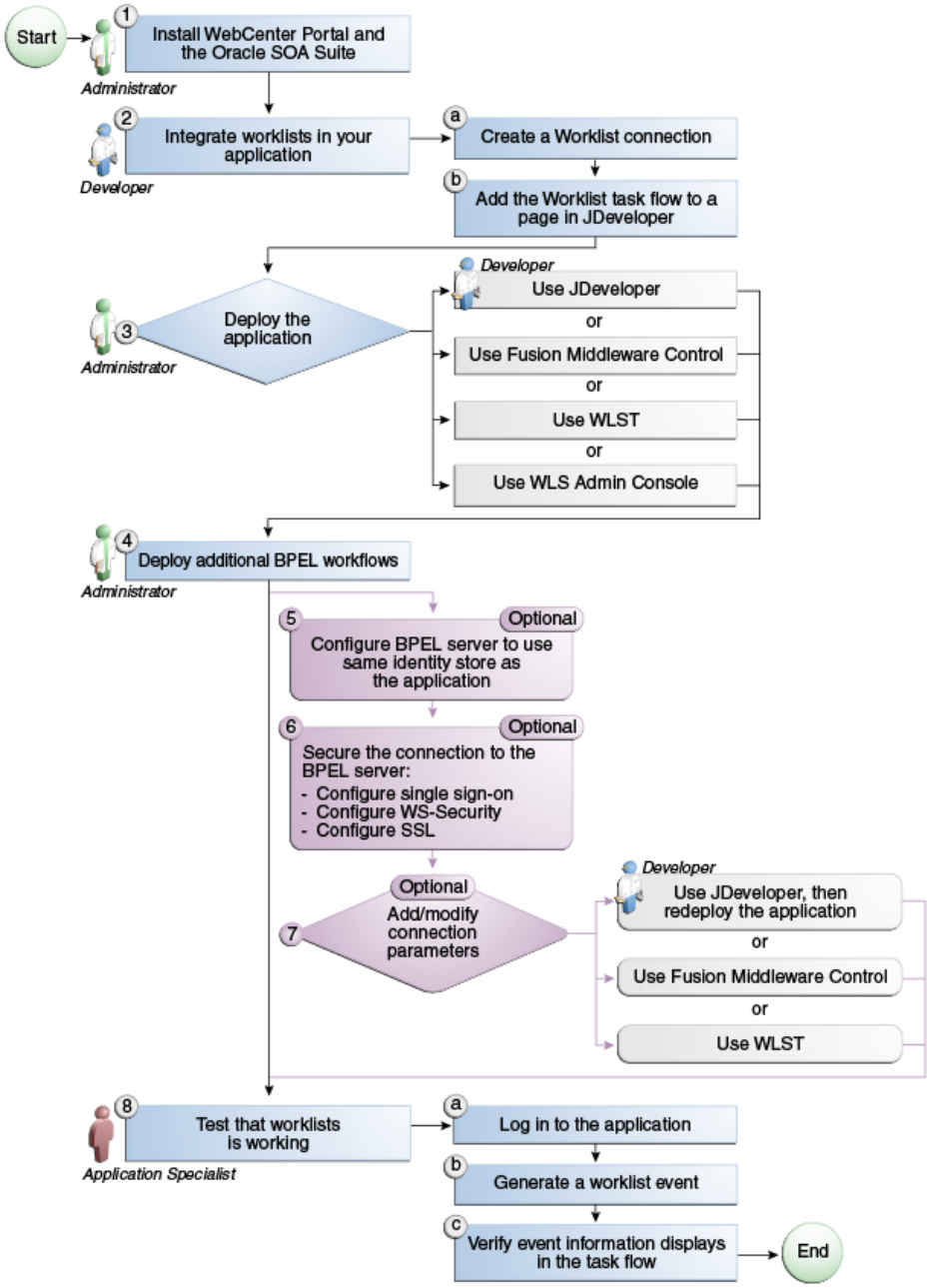


Table 20–2 Configuring Worklists for Portal Framework Applications

Actor	Task	Sub-Task
Administrator	**1.** Install WebCenter Portal and the Oracle SOA Suite	
Developer	**2.** Integrate worklist in your Portal Framework application	**2.a** Create a Worklist connection **2.b** Add the Worklist task flow to a page in JDeveloper

Table 20–2 (Cont.) Configuring Worklists for Portal Framework Applications

Actor	Task	Sub-Task
Developer/Administrator	**3.** Deploy the Portal Framework application using one of the following tools: ■ JDeveloper (Developer) ■ Fusion Middleware Control (Administrator) ■ WLST (Administrator) ■ WLS Admin Console (Administrator)	
Administrator	**4.** Deploy additional BPEL workflows	
Administrator	**5.** (Optional): Configure BPEL server to use same identity store as the application	
Administrator	**6.** (Optional): Secure the connection to the BPEL server	**6.a** Configure single sign-on **6.b** Configure WS-Security **6.c** Configure SSL
Developer/Administrator	**7.** (Optional): Add/modify connection parameters using one of the following tools: ■ JDeveloper, then redeploy the application (Developer) ■ Fusion Middleware Control (Administrator) ■ WLST (Administrator)	
End User	**8.** Test that worklist is working	**8.a** Log in to the Portal Framework application **8.b** Generate a worklist event **8.c** Verify event information displays in the task flow

20.2 About BPEL Connections

Consider the following while working with BPEL connections:

- Worklists allows multiple connections so that WebCenter Portal users can monitor and manage assignments and notifications from a range of BPEL servers. For more information, see Section 20.4, "Setting Up Worklist Connections."
- WebCenter Portal workflows require a single connection to the BPEL server included with the Oracle SOA Suite. For more information, see Section 20.5, "Specifying the BPEL Server Hosting WebCenter Portal Workflows."
- Worklists and the WebCenter Portal workflows can share the same BPEL server connection or each connect to different BPEL servers. To enable the display of worklist items created by the workflows in the current worklist of WebCenter Portal worklists, it is recommended that the WebCenter Portal workflows and worklist share a connection.
- Worklists can be wired to multiple BPEL connections to enable aggregation of worklist items from multiple BPEL servers. For example, when the topology

contains several BPEL servers running various workflow types, such as Human Resource and General Ledger servers.

- It is mandated that the BPEL connections are unique URLs. If this is not the case, then duplicate queries to the same server are created.

20.3 BPEL Server Prerequisites

Consider the following to ensure smooth functioning of worklists:

- Pages that include Worklist task flows must be secured through ADF security.
- Worklists must be configured to use an Oracle SOA Suite BPEL server that is accessible through the BPEL Worklists application. The URL is in the following format:

  `http://host:port/integration/worklistapp`

  If worklists is not running in the same domain as the Oracle SOA Suite BPEL server, then the identity store (LDAP) should be either shared (recommended) or contain identical user names.
- Clocks on the worklist's managed server and the Oracle SOA Suite BPEL's managed server must be synchronized such that the SAML authentication condition, `NotBefore`, which checks the freshness of the assertion, is not breached.
- No configuration-related exceptions must exist. Use the WLST command `listWorklistConnections` to display the configured connections and validate the connection details. After listing the connections, validate them using the URL property appended with `/integration/worklistapp`. Hence, verify that `http://host:port/integration/worklistapp` can access the BPEL Worklist application.
- If the Oracle SOA Suite BPEL's managed server is configured to use an identity store and that store does not contain `BPMWorkflowAdmin`, `weblogic` by default, then the `BPMWorkflowAdmin` user must be configured, as described in Section 20.7.2.2, "Shared User Directory Does Not Include the weblogic User."
- The `wsm-pm` application must be running on both worklists and Oracle SOA Suite's BPEL server's managed servers without any issues. This can be validated through the URL:

  `http://host:port/wsm-pm/validator`

For information on how to resolve BPEL server issues, see Section 20.7, "Troubleshooting Issues with Worklists."

This section includes the following subsections:

- Section 20.3.1, "BPEL Server - Installation and Configuration"
- Section 20.3.2, "BPEL Server - Security Considerations"
- Section 20.3.3, "BPEL Server - Limitations in WebCenter Portal"

20.3.1 BPEL Server - Installation and Configuration

Worklists relies on the Oracle BPEL Process Manager (BPEL) server, which is included with Oracle SOA Suite.

To work with worklist, you must install Oracle SOA Suite. For information about how to install Oracle SOA Suite, see the *Oracle Fusion Middleware Installation Guide for Oracle SOA Suite and Oracle Business Process Management Suite*.

After installing Oracle SOA Suite, you can integrate worklists into WebCenter Portal by setting up connections to the BPEL server.

20.3.2 BPEL Server - Security Considerations

Worklists display tasks for the currently authenticated user. For portal users to store and retrieve tasks on an Oracle SOA Suite BPEL server, their user names must either exist in a shared user directory (LDAP), or be set up similarly on both the BPEL Server and the portal application (WebCenter Portal or your own Portal Framework application).

For example, if the user `rsmith` wants to use worklist to store and retrieve tasks from the BPEL server, you must ensure that the user `rsmith` exists on both the BPEL server and within WebCenter Portal.

To access BPEL task details from the WebCenter Portal worklist component, without incurring additional login prompts, WebCenter Portal and Oracle SOA Suite servers must be configured to a shared Oracle Single Sign-On server. For more information, see Section 33.2, "Configuring Oracle Access Manager (OAM)" and Section 33.3, "Configuring Oracle Single Sign-On (OSSO)."

For a secure connection you can optionally configure WS-Security between SOA and WebCenter Portal. For information, see Chapter 36, "Configuring WS-Security."

20.3.3 BPEL Server - Limitations in WebCenter Portal

Worklist task flows function inside authenticated pages only. If Worklist task flows are placed on unsecured pages, that is public pages that are not navigated to from an application on which the user has logged in, the warning message "You must log in to view Worklist content" is displayed. This is done to ensure that a session for the current users is available to determine which user's tasks are to be queried.

20.4 Setting Up Worklist Connections

This section includes the following subsections:

- Section 20.4.1, "About Worklist Connections"
- Section 20.4.2, "Registering Worklist Connections"
- Section 20.4.3, "Activating a Worklist Connection"
- Section 20.4.4, "Modifying Worklist Connection Details"
- Section 20.4.5, "Deleting Worklist Connections"

20.4.1 About Worklist Connections

Worklists enables WebCenter Portal or your ownPortal Framework applications to show authenticated users a list of BPEL worklist items currently assigned to them. BPEL worklist items are open BPEL tasks from one or more BPEL worklist repositories.

A connection to every BPEL server that delivers worklist items is required. Multiple worklist connections are allowed so that WebCenter Portal users can monitor and manage assignments and notifications from a range of BPEL servers.

If a BPEL server cannot be contacted, then the Worklist task flow indicates that the connection is unavailable and any reason for the error is recorded in the server's diagnostic log. This log is located on the server that hosts the worklist component's log directory. For example:

`./user_projects/domains/base_domain/servers/WC_CustomPortal/logs/WC_CustomPortal-diagnostic.log`.

For a Portal Framework application:
`./user_projects/domains/base_domain/servers/WC_CustomPortal/logs/WC_CustomPortal-diagnostic.log`.

Note that these examples are for a domain named `base_domain`, which is the default, but can be defined differently during domain creation.

WebCenter Portal requires a BPEL server connection to support its internal workflows, that is, membership notifications and portal subscription requests. For more information, see Section 20.5, "Specifying the BPEL Server Hosting WebCenter Portal Workflows."

Worklists can share the SOA instance connection and by doing so, display worklist items relating to portal activity in the Worklist task flow of each user.

20.4.2 Registering Worklist Connections

This section includes the following subsections:

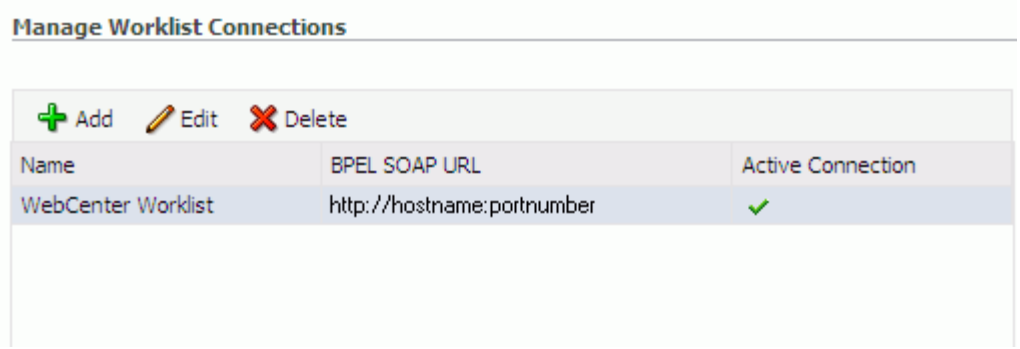
- Section 20.4.2.1, "Registering Worklist Connections Using Fusion Middleware Control"
- Section 20.4.2.2, "Registering Worklist Connections Using WLST"

20.4.2.1 Registering Worklist Connections Using Fusion Middleware Control

To register a Worklist connection:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal or the Portal Framework application. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.
3. On the **WebCenter Portal Service Configuration** page, select **Worklist**.
4. To register a new connection, click **Add** (Figure 20–3).

Figure 20–3 Configuring Worklist Connections



5. Enter a unique name for the Worklist connection and set it as the active connection (Table 20–3). This connection is picked up after you restart the managed server.

Table 20–3 Worklist Connection - Name

Field	Description
Connection Name	Enter a unique name for the connection. The name must be unique (across all connection types) within WebCenter Portal. This name may be displayed to users working with the worklist feature in WebCenter Portal. Users may organize their worklist assignments through various sorting and grouping options. The option "Group By Worklist Server" displays the name you specify here so it's important to enter a meaningful name that other users will easily recognize, for example, `Human Resources`.
Active Connection	Select to activate this worklist connection in WebCenter Portal. Once activated, worklist items from the associated BPEL server display in users' worklists. Multiple worklist connections may be active at a time, enabling WebCenter Portal users to monitor and manage assignments and notifications from a range of BPEL servers. If you need to disable a connection for any reason, deselect this option. (Edit mode only.) Check boxes indicate whether other components share this connection: ▪ **Worklist** Indicates whether the worklist displays items associated with this connection. ▪ Spaces Application Indicates whether WebCenter Portal uses the same BPEL server connection for internal workflows, such as membership notifications, subscription requests, and more. The BPEL server that provides this functionality is the BPEL server included with the Oracle SOA Suite. For more information, see Section 20.5, "Specifying the BPEL Server Hosting WebCenter Portal Workflows." Although not shown here, the notifications might be set up to use the BPEL server connection too. See Section 19.3, "Setting Up Notifications." Before modifying connection properties, consider the impact to any other components that share this connection.

6. Enter connection details for the BPEL server (Table 20–4).

Table 20–4 Worklist Connection - Connection Details

Field	Description
BPEL Soap URL	Enter the URL required to access the BPEL server. Use the format: `protocol://host:port` For example: `http://mybpelserver.com:8001` **Note:** WebCenter Portal uses the BPEL server included with the Oracle SOA Suite to implement WebCenter Portal workflows. If you are setting up the workflow connection, make sure you enter the SOA Suite's BPEL server URL here. For more information, see Section 20.5, "Specifying the BPEL Server Hosting WebCenter Portal Workflows."
SAML Token Policy URI	Select the SAML (Security Assertion Markup Language) token policy this connection uses for authentication. SAML is an XML-based standard for passing security tokens defining authentication and authorization rights. An attesting entity (that has a trusted relationship with the receiver) vouches for the verification of the subject by method called sender-vouches. Options available are: ■ **SAML Token Client Policy** (oracle/wss10_saml_token_client_policy) - Select to verify your basic configuration without any additional security. This is the default setting. ■ **SAML Token With Message Protection Client Policy** (oracle/wss10_saml_token_with_message_protection_client_policy) - Select to increase the security using SAML-based BPEL Web Services. If selected, you must configure keys stores both in WebCenter Portal and in the BPEL application. For information, see Chapter 36, "Configuring WS-Security." ■ Global Policy Attachment - Select this option when your environment supports Global Policy Attachment.
Recipient Key Alias	The recipient key alias to be used for message protected SAML policy authentication. Required only when the BPEL server connection is using a SAML token policy for authentication and WebCenter Portal's worklist is using multiple BPEL server connections. For example, `myKey` To determine the recipient key alias for a complex topology, see Section 36.3, "Configuring WS-Security for a Complex Topology."
Link URL	Specify the URL used to link to the BPEL server. Required only if it is different to the BPEL SOAP URL, for example, when SSO or HTTPS is configured. Use the format: `protocol://host:port` For example, `http://mySSO.host.com:7777` For performance reasons, in an HTTPS or SSO environment, the **Link URL** specifies user access to BPEL worklist items, through HTTPS or SSO Web servers, whereas the **BPEL SOAP URL** specifies direct access to BPEL Web services, without redirection through HTTPS or SSO Web servers.

7. Click **OK** to save this connection.

8. Click **Test** to verify if the connection you created works.

   For a successful connection, the Test Status message displays the advice that to start using the new (active) connection, you must restart the managed server on which the WebCenter Portal is deployed. For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

   See Section 20.7, "Troubleshooting Issues with Worklists" if the test fails.

   > **Tip:** To activate newly registered connections, perform the steps described in Section 20.4.3, "Activating a Worklist Connection."

20.4.2.2 Registering Worklist Connections Using WLST

Use the WLST command `createBPELConnection` to create a BPEL server connection. For command syntax and examples, see the "createBPELConnection" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

To configure a worklist to actively use a new BPEL server connection, some additional configuration is required. For more information, see Section 20.4.3.2, "Activating a Worklist Connections Using WLST."

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

> **Note:** To activate newly registered connections, perform the steps described in Section 20.4.3, "Activating a Worklist Connection."
>
> To start using the new (active) connection you must restart the managed server on which WebCenter Portal is deployed. For more information, see the "Starting and Stopping Managed Servers Using the Command Line" section in the *Oracle Fusion Middleware Administrator's Guide*.

20.4.3 Activating a Worklist Connection

In WebCenter Portal and Portal Framework applications, multiple Worklist connections may be active at a time. Multiple connections enable WebCenter Portal users to monitor and manage assignments and notifications from a multiple BPEL servers. From time to time you may need to temporarily disable an active connection so no errors or warnings are logged or displayed in the user interface, if the worklist queries a SOA server which is undergoing maintenance.

This section includes the following subsections:

- Section 20.4.3.1, "Activating a Worklist Connections Using Fusion Middleware Control"
- Section 20.4.3.2, "Activating a Worklist Connections Using WLST"

20.4.3.1 Activating a Worklist Connections Using Fusion Middleware Control

To activate or disable a Worklist connection:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal or the Portal Framework application. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"

- Section 6.3, "Navigating to the Home Page for Portal Framework Applications"

2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.
3. On the **WebCenter Portal Services Configuration** page, select **Worklist**.

   The Manage Worklist Connections table indicates currently active connections (if any).
4. Select the Worklist connection you want to activate (or disable), and then click **Edit**.
5. Select the **Worklist** check box to activate this Worklist connection in the WebCenter Portal.

   Once activated, worklist items from the associated BPEL server display in Worklist task flows. If you need to disable a connection for any reason, deselect this option.
6. Click **OK** to update the connection.
7. Click **Test** to verify if the connection you activated works.

   For a successfully activated connection, the Test Status message displays the advice that to start using the updated connection, you must restart the managed server on which WebCenter Portal is deployed. For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

20.4.3.2 Activating a Worklist Connections Using WLST

Use the WLST command `addWorklistConnection` to activate an existing BPEL connection for worklists. For command syntax and examples, see the "addWorklistConnection" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

To subsequently disable a BPEL connection used by worklists, run the WLST command `removeWorklistConnection`. Connection details are retained but the connection is no longer named as an active connection. For syntax details and examples, see the "removeWorklistConnection" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

Use `listWorklistConnections` to see which connections are currently active.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Note: To start using the active connection you must restart the managed server on which WebCenter Portal is deployed. For more information, see the "Starting and Stopping Managed Servers Using the Command Line" section in the *Oracle Fusion Middleware Administrator's Guide*.

20.4.4 Modifying Worklist Connection Details

This section includes the following subsections:

- Section 20.4.4.1, "Modifying Worklist Connection Details Using Fusion Middleware Control"
- Section 20.4.4.2, "Modifying Worklist Connection Details Using WLST"

20.4.4.1 Modifying Worklist Connection Details Using Fusion Middleware Control

To update worklist connection details:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal or the Portal Framework application. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.
3. On the **WebCenter Portal Services Configuration** page, select **Worklist**.
4. Select the Worklist connection you want to activate, and then click **Edit**.
5. Edit connection details, as required.

   For detailed parameter information, see Table 20–4, " Worklist Connection - Connection Details"
6. Click **OK** to update the connection.
7. Click **Test** to verify if the updated connection works.

   For a successfully updated connection, the Test Status message displays the advice that to start using the updated connection, you must restart the managed server on which the WebCenter Portal is deployed. For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

20.4.4.2 Modifying Worklist Connection Details Using WLST

Use the WLST command `setBPELConnection` to edit existing BPEL server connection details. For command syntax and examples, see the "setBPELConnection" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Note: To start using the updated (active) connection you must restart the managed server on which WebCenter Portal is deployed. For more information, see the "Starting and Stopping Managed Servers Using the Command Line" section in the *Oracle Fusion Middleware Administrator's Guide*.

20.4.5 Deleting Worklist Connections

Several WebCenter Portal components can share the same worklist connection, that is, worklists, notifications, and workflows. Before you delete a worklist connection,

navigate to the Application Configuration page in Fusion Middleware Control (**WebCenter Portal** > **Settings** > **Application Configuration**) to verify whether workflows and notifications are using the connection.

This section includes the following subsections:

- Section 20.4.5.1, "Deleting Worklist Connections Using Fusion Middleware Control"
- Section 20.4.5.2, "Deleting Worklist Connections Using WLST"

20.4.5.1 Deleting Worklist Connections Using Fusion Middleware Control

To delete a worklist connection:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal or the Portal Framework application. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.
3. On the **WebCenter Portal Services Configuration** page, select **Worklist**.
4. Select the Worklist connection you want to delete, and then click **Delete**.
5. To confirm, click **Yes**.
6. To effect this change you must restart the managed server on which the application is deployed. For more information, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

20.4.5.2 Deleting Worklist Connections Using WLST

Use the WLST command `deleteConnection` to remove a BPEL connection previously registered for worklist. For command syntax and examples, see the "deleteConnection" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

Use the WLST command `removeWorklistConnection` remove a BPEL server that is configured in `adf-config.xml`. Worklist no longer uses the connection specified but BPEL server connection details are retained in `connections.xml` for future use.

Use the WLST command `deleteConnection` to remove a BPEL server connection from `connections.xml`.

For command syntax and detailed examples, see the "removeWorklistConnection" and the "deleteConnection" sections in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Restart the managed server so that the changes can take place.

20.5 Specifying the BPEL Server Hosting WebCenter Portal Workflows

WebCenter Portal uses the BPEL server included with the Oracle SOA Suite to host internal workflows, such as space membership notifications, space subscription requests, and so on. To enable workflow functionality inside the WebCenter Portal application, a connection to this BPEL server is required.

> **Note:** WebCenter Portal workflows must be deployed on the SOA managed server that WebCenter Portal is configured to use. See also, the "Back-End Requirements for WebCenter Portal Workflows" chapter in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*.

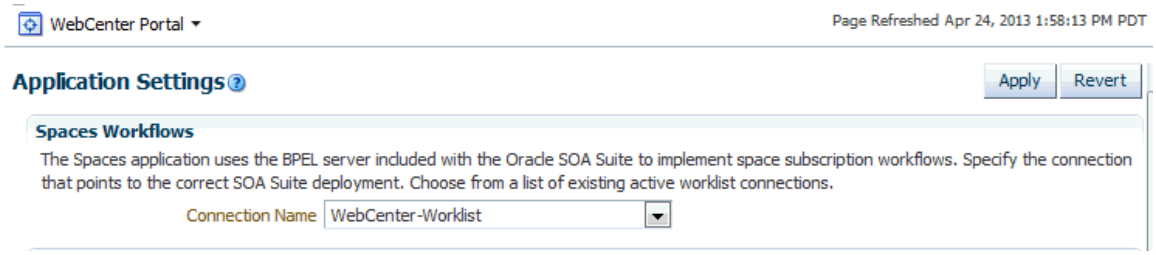
To configure a connection to WebCenter Portal workflows:

1. Login to Fusion Middleware Control, and navigate to the home page for WebCenter Portal.

   See Section 6.2, "Navigating to the Home Page for WebCenter Portal."

2. From the **WebCenter Portal** menu, select **Settings** > **Application Configuration**.

Figure 20–4 Choosing the BPEL Server Where WebCenter Portal Workflows are Deployed



3. From the **Connection Name** drop-down list, select the name of the connection you require.

   The connections on offer are those currently configured for the Worklist service in WebCenter Portal.

   Make sure that you select the connection that points to the SOA instance in which WebCenter Portal workflows are deployed. If that connection is not listed, you must create it. To define the connection, see Section 20.4, "Setting Up Worklist Connections."

4. Click **Apply**.

5. Restart WC_Spaces, the managed server on which the WebCenter Portal application is deployed, to effect this change.

   See Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

20.6 Configuring WebCenter Portal Workflow Notifications to be Sent by Email

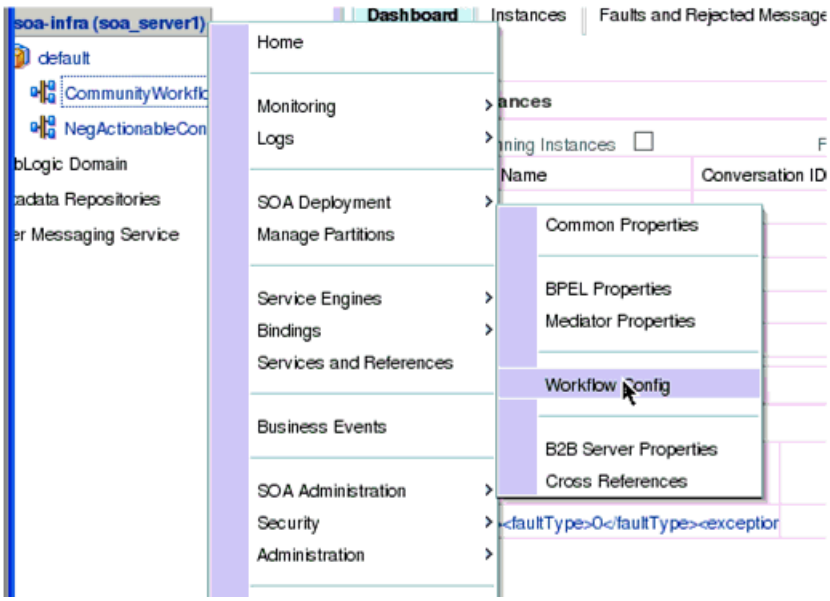
WebCenter Portal provides human workflows (requiring human interaction), which are integrated with SOA workflows. The SOA server can configure email so that

notifications are delivered to a user's inbox, where the user can accept or reject the notification.

This section describes how to enable email notifications and configure your mail server details to have WebCenter Portal workflow notifications sent to users by email.

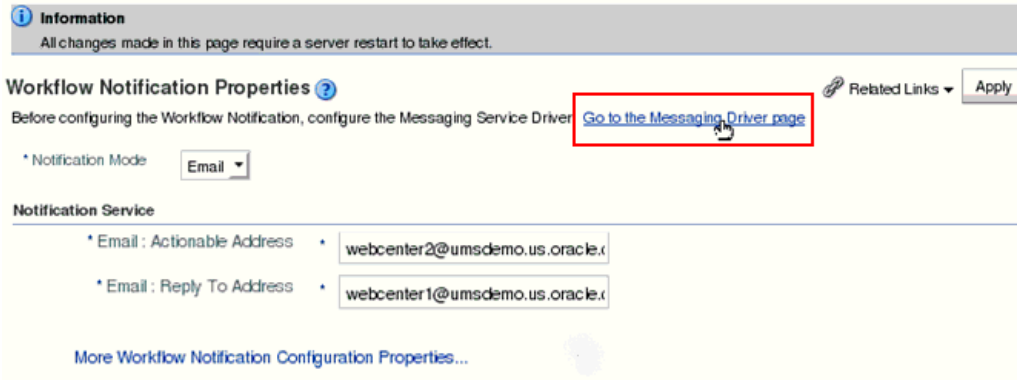
1. Use Fusion Middleware Control to update SOA to enable email notifications. Under the SOA server, select **SOA Administration**, then **Workflow Config**, as shown in Figure 20–5.

Figure 20–5 SOA Administration - Workflow Config



2. With **Email** selected as the **Notification Mode**, provide valid email accounts to use, (Figure 20–6).

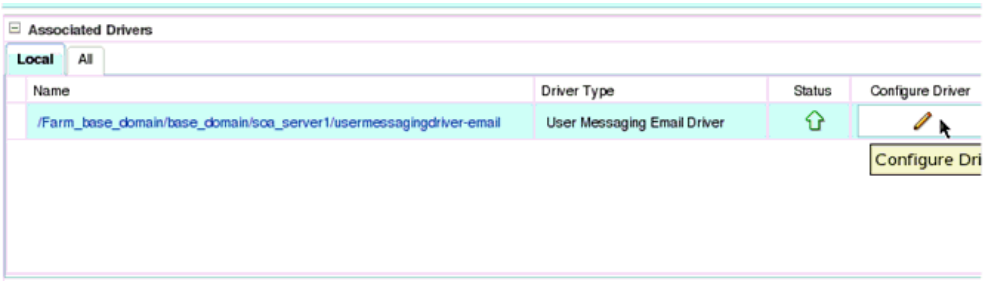
Figure 20–6 Email Notification Mode Properties



3. Click **Go to the Messaging Driver page** (Figure 20–6).

4. Select the **Configure Driver** icon for your User Messaging Email Driver (Figure 20–7).

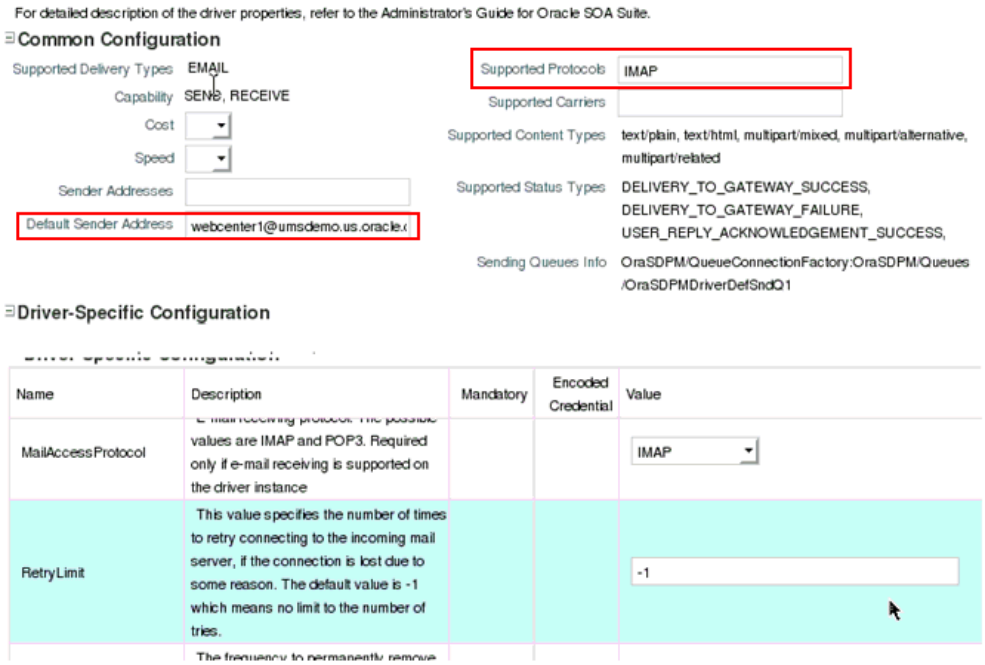
Figure 20–7 Associated Drivers



5. On the Configuration page for the email driver, do the following:

   - Under Common Configuration, enter your **Default Sender Address**. For **Supported Protocols** and **MailAccessProtocol**, enter IMAP (Figure 20–8).

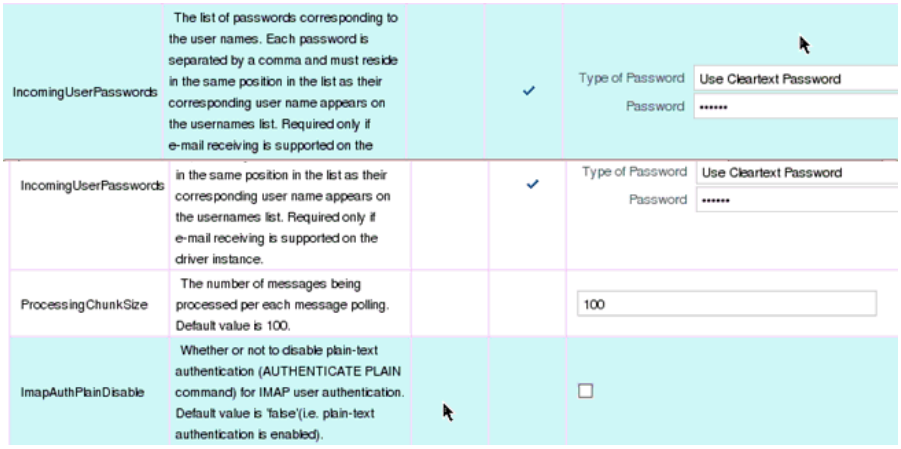
Figure 20–8 Configuration Page for the Email Driver



   - Under Driver-Specific Configuration, enter your email connection details, such as host, port, SSL, and incoming users/passwords, as shown in Figure 20–9 and Figure 20–10.

Figure 20–9 Driver-Specific Configuration

Figure 20–10 Driver-Specific Configuration, Continued



> **See Also:** "Configuring Oracle User Messaging Service" in *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite* for detailed information about driver properties and UMS configuration

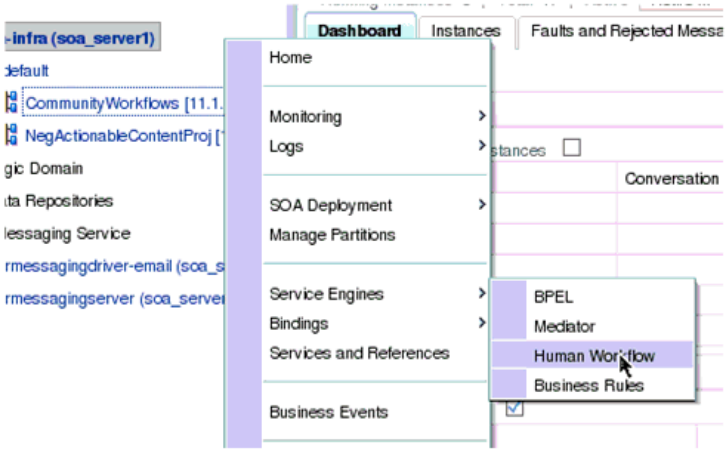
6. Save the configuration updates and restart the SOA managed server. (No configuration or restart is required for WebCenter Portal.)

   When a user is invited to join a portal, they are sent an email including **Accept** or **Reject** links to the invitation.

(**Optional**) You can validate that your configuration is correct by sending a sample email from Fusion Middleware Control.

7. From the Community Workflows, select Human Workflow (Figure 20–11)

Figure 20–11 Human Workflow

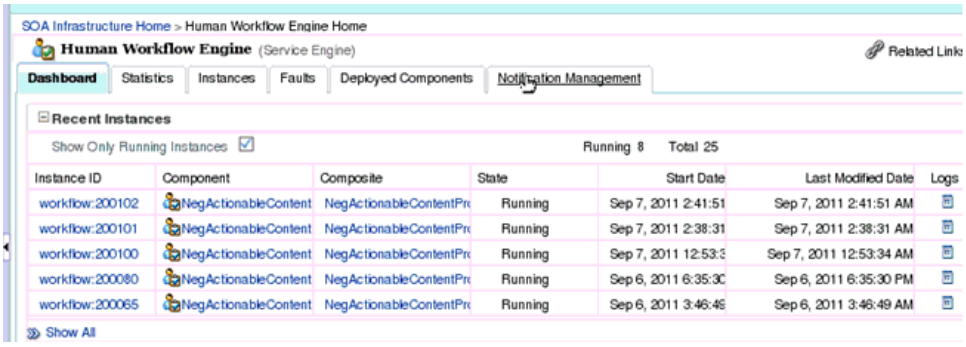


The Human Workflow Engine home page appears (Figure 20–12).

8. Review the email addresses in the **Notification Management** tab.

   The Notification Management tab provides options to view bad email addresses and resend notifications

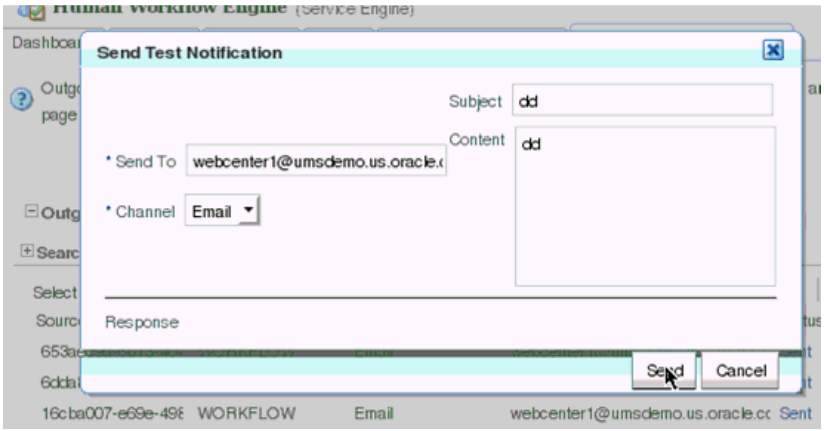
Figure 20–12 Human Workflow Engine Home



9. Click **Send Test Notification** to verify your configuration.

10. Click **Send** (Figure 20–13).

Figure 20–13 Test Notification



After sending the test notification, confirm that the email is received.

> **Note:** There are four workflows that generate an email where users can act upon the notification via email. In the portal administration settings **Members** page, you can add people and edit email notification messages. For more information, see the "Managing Members and Assigning Roles" section in *Oracle Fusion Middleware Administering Oracle WebCenter Portal*.

20.7 Troubleshooting Issues with Worklists

Worklists relies on several middleware components to display worklist items to logged-in users and therefore, several factors may cause worklists to fail. The issues and solutions discussed in this section relate to some common problems you may encounter.

> **See Also:** Section G.6, "Troubleshooting WebCenter Portal Workflows"

This section includes the following subsections:

- Section 20.7.1, "Unavailability of Worklists Due to Application Configuration Issues"
- Section 20.7.2, "Unavailability of Worklists Due to Server Failure"
- Section 20.7.3, "Email Notifications Not Working"

> **Note:** To identify causes of failures, examine log files on the managed servers hosting worklist processes and the managed servers for any SOA BPEL servers you have configured.

20.7.1 Unavailability of Worklists Due to Application Configuration Issues

Issues described in this section pertain to the unavailability of worklist—Worklist task flows display the message **The Worklist service is unavailable** with the following warning:

```
Either no BPEL connections are configured, or there is an issue
with the existing connection configuration. Verify that at least
one BPEL Worklist connection is configured for this application,
and that no unresolved "ConfigurationExceptions" exceptions are
logged.
```

This section includes the following subsections:

- Section 20.7.1.1, "adf-config.xml Refers to a Non-Existent BPEL Connection"
- Section 20.7.1.2, "adf-config.xml Has No Reference to a BPEL Connection"
- Section 20.7.1.3, "No Rows Yet Message Displays"

20.7.1.1 adf-config.xml Refers to a Non-Existent BPEL Connection

Problem

The connection listed in the `adf-config.xml` file does not exist in the application's `connections.xml` file. The following entries exist in the diagnostic log file for the managed server on which the application is running:

```
[2009-03-22T13:33:54.140+00:00] [DefaultServer] [WARNING]
[WCS-32008] [oracle.webcenter.worklist.config][tid:
[ACTIVE].ExecuteThread: '12' for queue: 'weblogic.kernel.Default
(self-tuning)'] userId: user][ ecid:
0000I0iOmdTFk3FLN2o2ye19kTB0000V,0][APP: Worklist#V2.0 arg:
Human Resources The BPEL Connection named'connection_name'was
not present in the connections.xml file. This will prevent the
Worklist service from being able to interact with the required
this BPEL connection.
```

Solution

Either create a BPEL connection with the name stated in the log, or remove the connection. For more information about how to update the worklist configuration post deployment, see Section 20.4, "Setting Up Worklist Connections."

During development, see the "Integrating Worklists" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

To find out which connections names are referenced and to validate the worklist configuration, run the WLST command, `listWorklistConnections(appName='myApp', verbose=true)`. For more information, see the "listWorklistConnections" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

20.7.1.2 adf-config.xml Has No Reference to a BPEL Connection

There is no reference to a worklist connection in the application's `adf-config.xml`, but this connection exists in the `connections.xml` file.

Problem

In diagnostic log files for the managed server on which the application is running, you see entries such as the following:

```
[2009-03-23T10:23:56.943+00:00] [DefaultServer] [WARNING]
[WCS-32009] [oracle.webcenter.worklist.config] [tid:
[ACTIVE].ExecuteThread: '21' for queue: 'weblogic.kernel.Default
(self-tuning)'] [userId: user] [ecid:
```

```
0000I0mqx8Fk3FLN2o2ye19lqBV000008,0] [APP: Worklist#V2.0] The
Worklist service does not have a ConnectionName configuration
entry in adf-config.xml that maps to a BPELConnection in
connections.xml, therefore the Worklist service was not
configured for this application.
```

Solution

Configure a connection to at least one BPEL server so that the worklist can query worklist items.

Post deployment, create Worklist connections through WLST or Fusion Middleware Control. For information, see Section 20.4.2, "Registering Worklist Connections." During development, create Worklist connections through Oracle JDeveloper. For information, see the "Integrating Worklists" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*. Do not modify `adf-config.xml` and `connections.xml` files manually.

20.7.1.3 No Rows Yet Message Displays

Problem

The Worklist task flow continues to display the **No Rows Yet** message.

Solution

The following are possible solutions to address this problem:

- No '**Assigned**' worklist items exist for the logged in user:

  If worklist items are assigned to the logged-in user and the state of these items is **Assigned**, then they always show in the Worklist task flow. The **No Rows Yet** message indicates that no assigned Worklist items exist for the logged-in user. This is not an issue, but expected behavior.

  To confirm that this message is displaying correct information, open the Oracle SOA Suite BPEL Worklist application, and check whether any worklist items exist. The URL of BPEL Worklist application is: `http://host:port/integration/worklistapp`. Where `host` and `port` are the same as those used in the Worklist connection.

- The ADF page on which the Worklist task flow exists is not ADF-secured:

  The Worklist task flow is not able to query the Worklist repository, because there is no authenticated user associated with the application session to access the Oracle SOA Suite BPEL server. Apply the ADF security on the page. For information, see the "Setting Security for Worklists" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

20.7.2 Unavailability of Worklists Due to Server Failure

Server failure is the likely cause of an issue if a worklist connection exists, and the Worklist task flow shows the **The Worklist service is unavailable** warning. In case of multiple connections, the **More items not currently available** message displays. These generic warning messages display when there is an issue with worklist interactions with the Oracle SOA Suite BPEL repository.

To identify the root cause of the issue, examine the managed server's diagnostic logs at the time when it fails. In some cases, it is necessary to also examine the log files of the managed server on which the Oracle SOA Suite BPEL processes run. Typically, an

entry such as the following exists in diagnostic logs of the worklist application's managed server:

```
[2009-03-23T11:35:21.735+00:00] [DefaultServer] [ERROR]
[WCS-32100] [oracle.webcenter.worklist.model] [tid:
[ACTIVE].ExecuteThread: '0' for queue: 'weblogic.kernel.Default
(self-tuning)'] [userId: user] [ecid:
0000I0n7GBZFk3FLN2o2ye19lrBX00000L,0] [APP: Worklist#V2.0] [arg:
WebCenter Worklist] The WebCenter Worklist has queried the BPEL
Worklist connection named 'WebCenter Worklist', and encountered
a WebCenter Executor error. Please see related exception for
details. If the WebCenter Worklist is running in an Application
Server, check to see if the wsm-pm application is up and
running.
```

This states that there is an issue with the `wsm-pm` application that is used for WS security. There can also be some other causes related to the exception. It is recommended that you examine the logged exceptions on both the WebCenter managed server and the configured Oracle SOA suites managed servers when these issues occur.

This section includes the following sub sections:

- Section 20.7.2.1, "Users Mismatch in Identity Stores"
- Section 20.7.2.2, "Shared User Directory Does Not Include the weblogic User"
- Section 20.7.2.3, "Issues with the wsm-pm Application"
- Section 20.7.2.4, "Clocks are Out of Sync for More Than Five Minutes"
- Section 20.7.2.5, "Worklist Timed Out or is Disabled"

20.7.2.1 Users Mismatch in Identity Stores

Mismatch in identity stores used by the managed server on which the Worklist task flow is running and that of the Oracle SOA Suite BPEL server.

Problem

If a user exists in the worklist managed server's identity store but not in the Oracle SOA Suite's identity store, then the following messages display:

In the diagnostic logs of the worklist's managed server:

```
[2009-03-23T11:35:21.407+00:00] [DefaultServer] [ERROR] []
[oracle.webcenter.worklist.config] [tid: pool-1-daemon-thread-12] [userId: Luke]
[ecid: 0000I0n7GBZFk3FLN2o2ye19lrBX00000L,0:1:3] [APP: Worklist#V2.0] Error in
workflow service Web service operation invocation.[[
Error in workflow service Web service operation invocation. The error is .
Verify that the SOAP connection information for the server is correct.
 ORABPEL-30044
Error in workflow service Web service operation invocation.
Error in workflow service Web service operation invocation. The error is .
Verify that the SOAP connection information for the server is correct.
    at
oracle.bpel.services.workflow.query.client.TaskQueryServiceSOAPClient.convertSOAPF
aultException(TaskQueryServiceSOAPClient.java:242)
    at
oracle.bpel.services.workflow.query.client.TaskQueryServiceSOAPClient.invoke(TaskQ
ueryServiceSOAPClient.java:203)
    at
oracle.bpel.services.workflow.query.client.TaskQueryServiceSOAPClient.authenticate
```

```
(TaskQueryServiceSOAPClient.java:253)
    at
oracle.bpel.services.workflow.query.client.AbstractDOMTaskQueryServiceClient.authe
nticate(AbstractDOMTaskQueryServiceClient.java:164)
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at
sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:39)
    at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:
25)
    at java.lang.reflect.Method.invoke(Method.java:597)
    at oracle.webcenter.concurrent.MethodTask.call(MethodTask.java:34)
    at oracle.webcenter.concurrent.Submission$2.run(Submission.java:492)
    at java.security.AccessController.doPrivileged(Native Method)
    at oracle.security.jps.util.JpsSubject.doAsPrivileged(JpsSubject.java:313)
    at oracle.webcenter.concurrent.Submission.runAsPrivileged(Submission.java:499)
    at oracle.webcenter.concurrent.Submission.run(Submission.java:433)
    at
oracle.webcenter.concurrent.Submission$SubmissionFutureTask.run(Submission.java:77
9)
    at java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:441)
    at java.util.concurrent.FutureTask$Sync.innerRun(FutureTask.java:303)
    at java.util.concurrent.FutureTask.run(FutureTask.java:138)
    at
oracle.webcenter.concurrent.ModifiedThreadPoolExecutor$Worker.runTask(ModifiedThre
adPoolExecutor.java:657)
    at
oracle.webcenter.concurrent.ModifiedThreadPoolExecutor$Worker.run(ModifiedThreadPo
olExecutor.java:682)
    at java.lang.Thread.run(Thread.java:619)
]]
[2009-03-23T11:35:21.735+00:00] [DefaultServer] [NOTIFICATION] []
[oracle.webcenter.worklist.config] [tid: pool-1-daemon-thread-15] [userId: Luke]
[ecid: 0000I0n7GBZFk3FLN2o2ye19lrBX00000L,0:1:6] [APP: Worklist#V2.0]
TaskServiceSOAPClient: soapFault:[[
<env:Fault
xmlns:ns0="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-sece
xt-1.0.xsd"xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
   <faultcode>ns0:FailedAuthentication</faultcode>
   <faultstring>FailedAuthentication : The security token cannot be authenticated
or authorized.</faultstring>
   <faultactor/>
</env:Fault>
]]
```

In the diagnostic logs of the Oracle SOA Suite's managed server:

```
[2009-03-23T04:52:07.909-07:00] [soa_server1] [ERROR]
[WSM-00008] [oracle.wsm.resources.security] [tid:
[ACTIVE].ExecuteThread: '2' for queue: 'weblogic.kernel.Default
(self-tuning)'] [userId: <anonymous>] [ecid:
0000I0nB64fFk3FLN2o2ye19lrBX00000O,0:1:3:1]
[WEBSERVICE_PORT.name: TaskQueryServicePortSAML] [APP:
soa-infra] [J2EE_MODULE.name:
/integration/services/TaskQueryService] [WEBSERVICE.name:
TaskQueryService] [J2EE_APP.name: soa-infra] Web service
authentication failed.
```

Solution

The same users must exist in identity stores of both managed servers. For information, see the "Setting Security for Worklists" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

This can be easily accomplished with a common LDAP identity store. A useful check is to validate that you can log in to the Oracle SOA Suite's BPEL worklist application with the user ID for which the worklist is unavailable. That is, try accessing the integration worklist application at: `http://host:port/integration/worklistapp`. Where the `host` and `port` are the same as those used in the worklist connection for the task flow application.

20.7.2.2 Shared User Directory Does Not Include the weblogic User

Problem

BPEL Web services cannot respond to requests received from the worklist because the shared user directory does not include the `weblogic` user.

Solution

Ensure that you have tried the solution provided in Users Mismatch in Identity Stores. If that solution did not resolve the issue, then try the solution described in this section.

If Oracle SOA Suite is connected to a shared user directory (LDAP), and the user `weblogic` does not exist in the identity store, then the following step assigns the `BPMWorkflowAdmin` role to a valid user in the identity store. Use WLST to revoke an application role from `SOAAdmin` and grant it to a member of the external identity store. This can be done by running the following WLST command from the *SOA_ORACLE_HOME*. For example:

```
cd SOA_ORACLE_HOME/common/bin/
wlst.sh
connect('weblogic','weblogic', '## soa host ##:## soa administration port ##')
revokeAppRole(appStripe="soa-infra", appRoleName="BPMWorkflowAdmin",
     principalClass="oracle.security.jps.service.policystore.ApplicationRole",
principalName="SOAAdmin")
grantAppRole(appStripe="soa-infra", appRoleName="BPMWorkflowAdmin",
     principalClass="weblogic.security.principal.WLSUserImpl",
principalName="user")
```

In this example, the LDAP identity store has a user named `user`. If the user to which you want to grant the `BPMWorkflowAdmin` role does not exist in the LDAP identity store, then you must restart the Oracle SOA Suite's managed server to make this change effective.

20.7.2.3 Issues with the wsm-pm Application

Problem

Issue with the `wsm-pm` application on either the worklist's managed server, or the Oracle SOA Suite's managed server, or on both.

Solution

The `wsm-pm` application manages the Web service security policies that control the SAML authentication in the worklist. To validate the `wsm-pm` application, log in to the `wsm-pm` application's validation page as a user with administrative rights. Use this format for validation: `http://host:port/wsm-pm/validator`. If there are no

issues with this application, then accessible policies must display. If policies do not display, then investigate the related logged information on the server whose `wsm-pm` application is failing.

20.7.2.4 Clocks are Out of Sync for More Than Five Minutes

Due to security reasons, the Web service security interaction between the worklist's managed server and that of the Oracle SOA Suite BPEL must take place with a time difference of less than five minutes. That is, the clocks on both host machines must have a time difference of less than five minutes, otherwise authentication fails. The SAML assertion uses the `NotBefore` condition to verify this.

Problem

Clocks of the worklist's managed server and the Oracle SOA Suite BPEL's managed server are out of sync for more than five minutes.

Solution

Ensure that the current time is not set to earlier than the SAML assertion's `clockskew`, which is 300 seconds by default.

Either match the time on the machines, or configure the `agent.clock.skew` property (in seconds) in the `policy-accessor-config.xml` file. This file is located in the *`DOMAIN_HOME`*`/config/fmwconfig` directory.

20.7.2.5 Worklist Timed Out or is Disabled

Problem

The worklist cannot obtain a query result from the Oracle SOA Suite BPEL server within a defined period.

The worklist issues queries to the Oracle SOA Suite BPEL server using concurrent threads. These threads are allotted a certain amount of time in which to respond. If these threads do not respond in the allotted time, for example 15 seconds, then the worklist times out the call, and it allows the task flow to display the unavailability message. In such a case, log files include related exceptions such as the following:

```
[2009-03-03T12:09:34.769-08:00] [WLS_Spaces] [ERROR] [WCS-32103]
[oracle.webcenter.worklist.model] [tid: [ACTIVE].ExecuteThread: '3' for queue:
'weblogic.kernel.Default (self-tuning)'] [userId: user] [ecid:
0000HzDx68KC0zT6uBbAEH19fOWs00002q,0] [APP: webcenter] Unable to query BPEL
repository.[[
oracle.webcenter.concurrent.TimeoutException: Execution timedout
      queued :     1 ms
   suspended :     0 ms
     running : 15389 ms
     timeout : 15000 ms
     service : Worklist
    resource : ir
      source : oracle.webcenter.concurrent.CallableTask@bf3952
(oracle.webcenter.concurrent.CallableTask)
  submission : 150
        at
oracle.webcenter.concurrent.Submission.transitionTo(Submission.java:595)
        at oracle.webcenter.concurrent.Submission.timeout(Submission.java:634)
        at
oracle.webcenter.concurrent.InternalExecutorService.checkForTimeouts(InternalExecu
torService.java:566)
        at
```

```
oracle.webcenter.concurrent.InternalExecutorService.access$300(InternalExecutorSer
vice.java:18)
        at
oracle.webcenter.concurrent.InternalExecutorService$1.run(InternalExecutorService.
java:352)
        at java.util.TimerThread.mainLoop(Timer.java:512)
        at java.util.TimerThread.run(Timer.java:462)]]
```

Solution

If errors such as this occur consistently, then there may be fundamental issues with the resources available to the managed servers running the worklist and the Oracle SOA Suite BPEL server.

Validate that the volume of users and resources provided is adequate to run these servers in the infrastructure provided.

If you are unable to improve the SOA server's performance, then increase the timeout threshold using the Enterprise Manager System MBean Browser (adding a new timeout for the "Worklist").

For details, see the "Configuring Concurrency Management" section in the *Oracle Fusion Middleware Performance and Tuning Guide*.

Note: Continuous occurrence of `TimeoutExceptions` can also disable the worklist. Due to which it cannot connect to the BPEL instance that is failing to respond quickly. In such a case, the logs contain `oracle.webcenter.concurrent.DisabledException` exceptions. These exceptions are related to the worklist failure.

20.7.3 Email Notifications Not Working

Problem

Notifications for workflows are not being sent by email, as described in Section 20.6, "Configuring WebCenter Portal Workflow Notifications to be Sent by Email."

Solution

Check the error logs on the WebCenter and SOA servers for errors at the time when the invite process is instigated. If there appears to be an issue with the email configuration, then validate that you can use the exact same LDAP settings and user accounts to send and receive emails using a different email client.

21

Managing Portlet Producers

This chapter describes how to create a connection to, or register, WSRP and Oracle PDK-Java portlet producers, and how to edit and delete those connections. This chapter also describes how to deploy WSRP and Oracle PDK-Java portlet producers.

This chapter includes the following topics:

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console and the `Administrator` role in the deployed application:

- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.
- **Portal Framework application**: `Administrator` role granted through the Administration Console.

For more information about roles and permissions, see Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

Note: Pagelet producer registration is described in a different chapter. For details, see Section 22.2, "Registering Pagelet Producer."

21.1 About Portlet Producers

System administrators can use Fusion Middleware Control or the WLST command-line tool to register and manage WSRP and Oracle PDK-Java portlet producers for WebCenter Portal application deployments.

Application administrators can also register and manage portlet producers at runtime through out-of-the-box administration pages or using the portlet producer task flow.

Consider the following while working with portlet producers:

- Some out-of-the-box producers are provided with WebCenter Portal: OmniPortlet and WSRP Tools. The following EAR files are packaged with WebCenter Portal:
  - `portalTools.ear` - OmniPortlet
  - `wsrp-tools.ear` - WSRP Tools

  You can install the `portalTools.ear` and `wsrp-tools.ear` files using the `registerOOTBProducers` WLST command. For command syntax and examples, see the "registerOOTBProducers" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.
- Before users can add JSR 286 or Oracle PDK-Java portlets to a page, you must register the owning WSRP and Oracle PDK-Java producers. See the "registerSampleProducers" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.
- The Oracle Portlet Producer product (server) must be installed in the production environment and the `wsrp-tools` and `portalTools` URLs must be accessible. If the Oracle Portlet Producer is not installed, see the "Extending an Existing Domain" section in the *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal* to install it in the production environment.
- When you create a connection to a portlet producer, the producer is registered with the WebCenter Portal application and the connection is added to the `connections.xml` file. For WRSP producers, a web service connection is also created, which follows the naming convention, *`connectionname-wsconn`*. For Oracle PDK-Java producers, an underlying URL connection is created, which follows the naming convention, *`connectionname-urlconn`*. During the registration, connection metadata is created in the Oracle Metadata Services (MDS) repository and in the producer being registered. When a producer's portlets are consumed, the user customizations are saved to the producer. During deregistration the producer connection and customizations are removed.
- All post deployment connection configuration is stored in MDS. For more information, see Section 1.3.5, "Oracle WebCenter Portal Configuration Considerations." For detailed information about MDS, see the "Managing the Metadata Repository" chapter in the *Oracle Fusion Middleware Administrator's Guide*.
- Portlet producer registration is dynamic. New portlet producers and updates to existing producers are immediately available in the WebCenter Portal application; it is not necessary to restart the WebCenter Portal application or the managed server.

- To migrate producers from one instance to another, use the migration utilities described in the "Migrating a WSRP Producer Persistence Store" or "Migrating a PDK-Java Producer Persistence Store" sections in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.
- For information on securing portlet producers, see Section 37.1, "Securing a WSRP Producer" and Section 37.2, "Securing a PDK-Java Producer."

21.2 Registering WSRP Producers

When you register a WSRP portlet producer, you provide basic information that describes the producer's operational parameters. This information is used by the application to communicate with the producer and with the portlets through the producer.

Oracle WebCenter Portal supports both WSPR 1.0 and WSRP 2.0 producers. The WSRP 2.0 standard provides support for, among others, interportlet communication and export and import of portlet customizations. You can leverage the benefits of WSRP 2.0 while building standard-based JSR 286 portlets.

Oracle WebCenter Portal provides several tools for registering WSRP producers with deployed applications.

This section includes the following topics:

- Section 21.2.1, "Registering a WSRP Producer Using Fusion Middleware Control"
- Section 21.2.2, "Registering a WSRP Producer Using WLST"
- Section 21.2.3, "Adding a Grant to the Policy Store for a Mapped User Identity"
- Section 21.2.4, "Registering a WSRP Portlet Producer in WebCenter Portal"
- Section 21.2.5, "Registering a WSRP Portlet Producer in WebCenter Portal Framework Applications"

21.2.1 Registering a WSRP Producer Using Fusion Middleware Control

To register a WSRP portlet producer:

1. Log in to Fusion Middleware Control and navigate to the home page for the Oracle WebCenter Portal application. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications".
2. Do one of the following:
   - For the WebCenter Portal application - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.
3. In the **Add Portlet Producer Connection** section, enter connection details for the WSRP producer.

   For detailed parameter information, see Table 21–1.

Table 21–1 WSRP Producer Connection Parameters

Field	Description
Connection Name	Enter a unique name to identify this portlet producer registration within the WebCenter Portal application. The name must be unique across all WebCenter Portal connection types. The name you specify here appears in Composer (under the *Portlets* folder).
Producer Type	Indicate the type of this producer. Select **WSRP Producer**.

WSDL URL	The registration URL for the WSRP producer.
	
	The syntax varies according to your WSRP implementation. For example, possible URL formats for a portlet deployed to the Oracle WSRP container include:
	
	`http://host_name:port_number/context_root/portlets/wsrp2?WSDL`
	
	`http://host_name:port_number/context_root/portlets/wsrp1?WSDL`
	
	`http://host_name:port_number/context_root/portlets/?WSDL` (WSRP 1.0 for backward compatibility)
	
	Where:
	
	■ `host_name` is the server where your producer is deployed.
	
	■ `port_number` is the HTTP listener port number.
	
	■ `context_root` is the Web application's context root.
	
	■ `portlets wsrp(1|2)?WSDL` is static text. All producers deployed to the Oracle WSRP container are exposed as WSRP version 1 and version 2 producers.
	
	In WebCenter Portal, only v2 WSDLs are supported for Oracle WebLogic Portal Producers.
	
	For example:
	
	`http://myhost.com:7778/MyPortletApp/portlets/wsrp2?WSDL`
	
	For WSRP producers, you can obtain this registration URL by accessing the producer test page at:
	
	`http://host_name:port_number/context_root/info`

Use Proxy?	Select if the WebCenter Portal application must use an HTTP proxy when contacting this producer. If selected, enter values for **Proxy Host** and **Proxy Port**. A proxy is required when the WebCenter Portal application and the remote portlet producer are separated by a firewall and an HTTP proxy is needed to communicate with the producer.
Proxy Host	Enter the host name or IP address of the proxy server. Do not prefix `http://` to the proxy server name.
Proxy Port	Enter the port number on which the proxy server listens. The default port is `80`.
Default Execution Timeout (Seconds)	Enter a suitable timeout for communications with the producer, in seconds. For example, the maximum time the producer may take to register, deregister, or display portlets on WebCenter Portal application pages. The default is 30 seconds. Individual portlets may define their own timeout period, which takes precedence over the value expressed here.

4. Use the **Security** section to specify the type of security token to use for the identity propagation/assertion.

   The security token with the propagated or asserted user information is represented as an XML element in the SOAP header. The security token and the SOAP message body are then digitally signed to prove the authenticity of the SOAP message origin from the WebCenter Portal application. WebCenter Portal applications support six types of security tokens: *WSS 1.0 Username Token Without Password*, *WSS 1.0 Username Token With Password*, *WSS 1.0 SAML Token*, *WSS 1.0 SAML Token With Message Integrity*, *WSS 1.0 SAML Token With Message Protection*, and *WSS 1.1 SAML Token With Message Protection*.

   Where SAML is an abbreviation for Security Assertion Markup Language.

   Note: PeopleSoft WSRP producers support two profiles: *Username Token With Password* and *SAML Token With Message Integrity*. Oracle Portal (as a consumer) supports three profiles: *Username Token Without Password*, *Username Token With Password*, *SAML Token With Message Integrity*. Other Oracle WSRP producers support all six profiles. For other WSRP containers, check with the specific vendor to determine the token formats they support.

   For detailed parameter information, see Table 21–2.

Table 21–2 WSRP Producer Security Connection Parameters

Field	Description
Token Profile	Select the type of token profile to use for authentication with this WSRP producer. Select from: ■ **WSS 1.0 SAML Token With Message Integrity** ■ **WSS 1.0 SAML Token With Message Protection** ■ **WSS 1.0 Username Token Without Password** ■ **WSS 1.0 Username Token With Password** ■ **WSS 1.0 SAML Token** ■ **WSS 1.1 SAML Token with Message Protection** ■ **None** For a description of each of these options, see Table 21–3.
Configuration	Select: ■ **Default** to use a default token profile configuration. ■ **Custom** to provide a custom Oracle Web Service Manager configuration. Additional security options display (including all the keystore properties) when you select **Custom**.
Issuer Name	Enter the name of the issuer of the SAML Token. For example: `www.example.com` The issuer name is the attesting entity that vouches for the verification of the subject, and it must be a trusted SAML issuer on the producer end. Valid for: WSS 1.0 SAML Token With Message Integrity, WSS 1.0 SAML Token With Message Protection. WSS 1.0 SAML Token, WSS 1.1 SAML Token with Message Protection

Table 21–2 (Cont.) WSRP Producer Security Connection Parameters

Field	Description
Default User	Enter a user name to assert to the remote producer when the user is not authenticated with the WebCenter Portal application. When unauthenticated, the identity *anonymous* is associated with the application user. The value *anonymous* may be inappropriate for the remote producer, so it may be necessary to specify an alternative identity here. Keep in mind though, that in this case, the WebCenter Portal application has not authenticated the user so the default user you specify should be a low privileged user in the remote producer. If the user has authenticated to the application, the user's identity is asserted rather than the default user. The remote WSRP producer must be set up to accept this information. You must also add a grant to the policy store as described in Section 21.2.3, "Adding a Grant to the Policy Store for a Mapped User Identity." Valid for: WSS 1.0 SAML Token With Message Integrity, WSS 1.0 SAML Token With Message Protection, WSS 1.0 SAML Token, WSS 1.1 SAML Token with Message Protection and WSS 1.0 Username Without Password.
Associated External Application (Username With Password)	If this producer uses an external application for authentication, use the **Associated External Application** drop-down list to identify the application. If the application you want is not listed, select **Create New** to define the external application now. An external application is required to support producers using the security option *WSS 1.0 Username With Password*. The external application stores and supplies the user credentials. See also Section 23.2, "Registering External Applications." Valid for: WSS 1.0 Username With Password only.

Table 21–3 lists the security token types supported by WebCenter Portal applications.

Table 21–3 Token Profiles Options

Token Profile	Description
WSS 1.0 SAML Token With Message Integrity `wss10_saml_token_with_message_integrity_client_policy`	This policy provides message-level integrity protection and SAML-based authentication for outbound SOAP requests in accordance with the WS-Security 1.0 standard. A SAML token, included in the SOAP message, is used in SAML-based authentication with sender vouches confirmation. This policy uses WS-Security's Basic 128 suite of asymmetric key technologies and SHA-1 hashing algorithm for message integrity.
WSS 1.0 SAML Token With Message Protection `oracle/wss10_saml_token_with_message_protection_client_policy`	This policy provides message-level protection (integrity and confidentiality) and SAML-based authentication for outbound SOAP requests in accordance with the WS-Security 1.0 standard. The web service consumer includes a SAML token in the SOAP header and the confirmation type is sender-vouches. This policy uses WS-Security's Basic 128 suite of asymmetric key technologies. Specifically, RSA key mechanisms for message confidentiality, SHA-1 hashing algorithm for message integrity, and AES-128 bit encryption.

Table 21–3 (Cont.) Token Profiles Options

Token Profile	Description
WSS 1.0 Username Token Without Password `oracle/wss10_username_id_propagation_with_msg_protection_client_policy`	This policy provides username (with password) token profile based identity propagation with certificate based message protection for outbound SOAP requests in accordance with the WS-Security 1.0 standard. Credentials (username only) are included in outbound SOAP request messages through a WS-Security UsernameToken header. No password is included. Message protection is provided using WS-Security 1.0's Basic 128 suite of asymmetric key technologies. Specifically, RSA key mechanisms for message confidentiality, SHA-1 hashing algorithm for message integrity, and AES-128 bit encryption.
WSS 1.0 Username Token With Password `oracle/wss10_username_token_with_message_protection_client_policy`	This policy provides username (with password) token profile based identity propagation with certificate based message protection for outbound SOAP requests in accordance with the WS-Security v1.0 standard. Both plain text and digest mechanisms are supported. This policy uses WS-Security's Basic 128 suite of asymmetric key technologies. Specifically, RSA key mechanism for message confidentiality, SHA-1 hashing algorithm for message integrity, and AES-128 bit encryption. Use this token profile if the WSRP producer has a different identity store. You will need to define an external application pertaining to the producer and associate the external application with this producer.
WSS 1.0 SAML Token `oracle/wss10_saml_token_client_policy`	This policy provides SAML-based authentication for outbound SOAP request messages in accordance with the WS-Security 1.0 standard. The policy propagates user identity and is typically used in intra departmental deployments where message protection and integrity checks are not required. This policy does not require any keystore configuration.
WSS 1.1 SAML Token with Message Protection `oracle/wss11_saml_token_with_message_protection_client_policy`	This policy provides message-level protection (integrity and confidentiality) and SAML token population for outbound SOAP requests in accordance with the WS-Security 1.1 standard. A SAML token, included in the SOAP message, is used in SAML-based authentication with sender vouches confirmation. This policy uses the symmetric key technology for signing and encryption, and WS-Security's Basic 128 suite of asymmetric key technologies for endorsing signatures.
None	No token. If None is selected, no WS-Security header is attached to the SOAP message.

5. Use the **Keystore** section to specify the location of the key store that contains the certificate and private key that is used for signing some parts (security token and SOAP message body) of the SOAP message.

   Only configure these properties if you want to override the configuration specified for the domain

   For detailed parameter information, see Table 21–4.

Table 21–4 WSRP Producer Key Store Connection Parameters

Field	Description
Recipient Alias	Specify the key store alias that is associated with the producer's certificate. This certificate is used to encrypt the message to the producer.

Table 21–4 (Cont.) WSRP Producer Key Store Connection Parameters

Field	Description
Store Path	Enter the absolute path to the keystore that contains the certificate and the private key that is used for signing or encrypting the SOAP message (security token and message body). The signature, encryption, and recipient keys described in this table must be available in this keystore. The keystore file specified must be created using JDK's keytool utility.
Password	Provide the password to the keystore that was set when the keystore was created. The producer is not available if a password is not specified or incorrect.
Signature Key Alias	Enter the signature key alias. The **Signature Key Alias** is the identifier for the certificate associated with the private key that is used for signing.
Signature Key Password	Enter the password for accessing the key identified by the alias specified in **Signature Key Alias**.
Encryption Key Alias	Enter the key alias used by the producer to encrypt the return message. A valid value is one of the key aliases that is located in the specified key store. This property is optional. If not specified, the producer uses the signing key for encrypting the return message.
Encryption Key Password	Enter the password for accessing the encryption key.

6. Click **OK**.

   The new producer appears in the connection table.

21.2.2 Registering a WSRP Producer Using WLST

Use the WLST command `registerWSRPProducer` to create a connection to a WSRP portlet producer and register the producer with your WebCenter Portal application. For command syntax and examples, see the "registerWSRPProducer" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

> **See Also:** `deregisterWSRPProducer`, `listWSRPProducers`, `refreshProducer`, `registerOOTBProducers`, `registerSampleProducers`

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

21.2.3 Adding a Grant to the Policy Store for a Mapped User Identity

If you are using the `Default User` field to map an alternative user identity you must also add a grant to the policy store by doing one of the following:

- Adding the following grant directly to the policy store:

```
<grant>
  <grantee>
   <codesource>

<url>file:${common.components.home}/modules/oracle.wsm.agent.common_11.1.1/wsm-
agent.jar</url>
```

```
   </codesource>
  </grantee>
  <permissions>
    <permission>
      <class>oracle.wsm.security.WSIdentityPermission</class>
      <name>resource=MyAppID</name>
      <actions>assert</actions>
    </permission>
  <permissions>
<grant>
```

Replacing ***MyAppID*** in the line above with the name of the client application, including the version number if any.

- Granting the permission by running the following WLST command:

```
grantPermission(codeBaseURL='file:${common.components.home}/modules/oracle.wsm.
agent.common_11.1.1/wsm-agent.jar',
permClass='oracle.wsm.security.WSIdentityPermission',
permTarget='resource=MyAppID', permActions='assert')
```

Replacing ***MyAppID*** with the name of the client application, including the version number if any.

21.2.4 Registering a WSRP Portlet Producer in WebCenter Portal

To register portlet producers in WebCenter Portal, you must have the following roles and permissions:

- **`AppConnectionManager` role**—Enables you to manage portlet producers.

  By default, users with the `Administrator` role in WebCenter Portal are assigned this role; and therefore, administrators can configure portlet producers. You can grant any other user this capability through Fusion Middleware Control or using the WLST command `grantAppRole`. For example, the following `grantAppRole` WLST command grants the `AppConnectionManager` role to the user `monty`:

  ```
  grantAppRole(appStripe='webcenter', appRoleName='AppConnectionManager',
  principalClass='weblogic.security.principal.WLSUserImpl',
  principalName='monty')
  ```

  See Section 32.6.2.1, "Granting Application Roles Using Fusion Middleware Control."

- **`Application - Manage Configuration` permission**—Enables access to the WebCenter Portal Administration pages. See Chapter 49, "Managing Security Across Portals."

To register a WSRP producer in WebCenter Portal:

1. Open WebCenter Portal Administration.

   For more information, see Section 47.2, "Accessing the Portal Builder Administration Page."

2. Click **Tools and Services**, and then select **Portlet Producers**.

   Alternatively, use the following URL, and then select Portlet Producers:

   `http://host:port/webcenter/portal/admin/tools`

3. On the menu bar, click **Register**.

4. In the Register Portlet Producer page, enter connection details for the WSRP portlet producer. For details, see Table 21–1.
5. Use the Security section to specify the type of security token to use for the identity propagation/assertion.

   The security token with the propagated or asserted user information is represented as an XML element in the SOAP header. The security token and the SOAP message body are then digitally signed to prove the authenticity of the SOAP message origin from the WebCenter Portal application.

   For details, see Table 21–5.

Table 21–5 WSRP Portlet Producer Registration - Security

Field	Description
Token Profile	Select the type of security token to use for the identity propagation or assertion. WebCenter Portal supports six types of security tokens: ▪ **WSS 1.0 SAML Token** ▪ **WSS 1.0 SAML Token With Message Integrity** ▪ **WSS 1.0 SAML Token With Message Protection** ▪ **WSS 1.0 Username Token With Password** ▪ **WSS 1.0 Username Token Without Password** ▪ **WSS 1.1 SAML Token With Message Protection** ▪ **None** For more information about each of these security tokens, see Table 21–3.
Recipient Alias	Specify the key store alias that is associated with the producer's certificate. This certificate is used to encrypt the message to the producer. Valid for: WSS 1.0 SAML Token With Message Protection, WSS 1.1 SAML Token with Message Protection and WSS 1.0 Username Without Password.
Default User	Enter a user name to assert to the remote producer when the user is not authenticated with the WebCenter Portal. When unauthenticated, the identity *anonymous* is associated with the application user. OWSM does not currently support the propagation of an *anonymous* identity, so you must specify an alternative identity here. Keep in mind though, that in this case, the WebCenter Portal has not authenticated the user so the default user you specify should be a low privileged user in the remote producer that is an appropriate identity to use for showing public content. For example, you may want to create a guest account in the identity store for this purpose. If the user has authenticated to the application, then the user's identity is asserted rather than the default user. The remote WSRP producer must be set up to accept this information. You must also add a grant to the policy store as described in Section 21.2.3, "Adding a Grant to the Policy Store for a Mapped User Identity." Valid for: WSS 1.0 SAML Token With Message Integrity, WSS 1.0 SAML Token With Message Protection, WSS 1.0 SAML Token, WSS 1.1 SAML Token with Message Protection and WSS 1.0 Username Without Password.

Table 21–5 (Cont.) WSRP Portlet Producer Registration - Security

Field	Description
Associated External Application	If this producer uses an external application for authentication, use the Associated External Application drop-down list to identify the application. An external application is required to support producers using the security option **WSS 1.0 Username Token With Password**. The external application stores and supplies the user credentials. See also Section 23.2, "Registering External Applications." Valid for: WSS 1.0 Username With Password only.

6. Click **Test** to verify that the server details you provided are correct.

   If the server is contactable, a success message is displayed. If the server is down or the host information is incorrect or no longer valid, a connection failure message is displayed.

   Note: The test performs a simple server (host/port) PING test. To verify whether the producer is accessible, access the producer's test page in your browser. For more information, see Section 21.3, "Testing WSRP Producer Connections."

7. Click **Ok**.

21.2.5 Registering a WSRP Portlet Producer in WebCenter Portal Framework Applications

For information about registering a WSRP portlet producer in Portal Framework applications at design time, see the "How to Register a WSRP Portlet Producer" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

For information about registering a WSRP portlet producer in Portal Framework applications at runtime, see Section 21.9.1, "Registering Portlet Producers with the Administration Console."

21.3 Testing WSRP Producer Connections

To verify a WSRP producer connection, first obtain the producer URL from:

`http://host_name:port_number/context_root/info`

Then, run the producer URL in a browser window.

For a WSRP v1 producer connection, the URL format is:

`http://host_name:port_number/context_root/portlets/wsrp1?WSDL`

For example:

```
http://myhost.com:7778/MyPortletApp/portlets/wsrp1?WSDL
```

For a WSRP v2 producer connection, the URL format is:

`http://host_name:port_number/context_root/portlets/wsrp2?WSDL`

For example:

```
http://myhost.com:7778/MyPortletApp/portlets/wsrp2?WSDL
```

21.4 Registering Oracle PDK-Java Producers

When you register a PDK-Java portlet producer, you provide basic information that describes the producer's operational parameters. This information is used by the WebCenter Portal to communicate with the producer and with the portlets through the producer.

WebCenter Portal provides several tools for registering Oracle PDK-Java producers with deployed applications.

This section includes the following topics:

- Section 21.4.1, "Registering an Oracle PDK-Java Producer Using Fusion Middleware Control"
- Section 21.4.2, "Registering an Oracle PDK-Java Producer Using WLST"
- Section 21.4.3, "Registering an Oracle PDK-Java Portlet Producer in WebCenter Portal"
- Section 21.4.4, "Registering an Oracle PDK-Java Portlet Producer in WebCenter Portal Framework Applications"

21.4.1 Registering an Oracle PDK-Java Producer Using Fusion Middleware Control

To register an Oracle PDK-Java portlet producer:

1. Log in to Fusion Middleware Control and navigate to the home page for the Oracle WebCenter Portal application. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For the WebCenter Portal application - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.
3. In the **Add Portlet Producer Connection** section, enter connection details for the Oracle PDK-Java producer.

   For detailed parameter information, see Table 21–6.

Table 21–6 Oracle PDK-Java Producer Connection Parameters

Field	Description
Connection Name	Enter a unique name that identifies this portlet producer registration within the WebCenter Portal application. The name must be unique across all WebCenter Portal connection types. The name you specify here appears in Composer (under the *Portlets* folder).
Producer Type	Indicate the type of this producer. Select **Oracle PDK-Java Producer**.

Table 21–6 (Cont.) Oracle PDK-Java Producer Connection Parameters

Field	Description

URL End Point	Enter the Oracle PDK-Java producer's URL using the following syntax:
	`http://host_name:port_number/context_root/providers`
	Where:
	▪ `host_name` is the server where the producer is deployed
	▪ `port_number` is the HTTP Listener port number
	▪ `context_root` is the Web application's context root
	▪ `providers` is static text
	For example:
	`http://myHost.com:7778/myEnterprisePortlets/providers`

Service ID	Enter a unique identifier for this producer.
	PDK-Java enables you to deploy multiple producers under a single adapter servlet. Producers are identified by their unique service ID. A service ID is required only if the service ID is not appended to the URL end point.
	For example, the following URL endpoint requires `sample` as the service ID:
	`http://domain.example.com:7778/axyz/providers`
	However, the following URL endpoint, does not require a service ID:
	`http://domain.example.com:7778/axyz/providers/sample`
	The service ID is used to look up a file called `<service_id>.properties`, which defines the characteristics of the producer, such as whether to display its test page. Use any value to create the service ID. When no Service ID is specified, `_default.properties` is used.

Use Proxy?	Select this check box if the WebCenter Portal application must use an HTTP proxy when contacting this producer. If selected, enter values for **Proxy Host** and **Proxy Port**. A proxy is required if the WebCenter Portal application and the remote portlet producer are separated by a firewall and an HTTP proxy is needed for communication with the producer.
Proxy Host	Enter the host name or IP address of the proxy server. Do not prefix `http://` to the proxy server name.
Proxy Port	Enter the port number on which the proxy server listens. The default port is `80`.
Associated External Application	If one of this producer's portlets requires authentication, use the **Associated External Application** drop-down to identify the correct external application. If the application you want is not listed, select **Create New** to define the external application now. See also Section 23.2, "Registering External Applications."

Table 21–6 (Cont.) Oracle PDK-Java Producer Connection Parameters

Field	Description
Establish Session?	Select to enable a user session when executing portlets from this producer. When sessions are enabled, they are maintained on the producer server. This allows the portlet code to maintain information in the session. Message authentication uses sessions, so if you specify a shared key, you must also select this option. For sessionless communication between the producer and the server, do not select this option.
Default Execution Timeout (Seconds)	Enter a suitable timeout for communications with the producer, in seconds. For example, the maximum time the producer may take to register, deregister, or display portlets on WebCenter Portal application pages. This defaults to `30` seconds. Individual portlets may define their own timeout period, which takes precedence over the value expressed here.
Subscriber ID	Enter a string to identify the consumer of the producer being registered. When a producer is registered with an application, a call is made to the producer. During the call, the consumer (WebCenter Portal application in this instance) passes the value for Subscriber ID to the producer. If the producer does not see the expected value for Subscriber ID, it might reject the registration call.
Shared Key	Enter a shared key to use for producers that are set up to handle encryption. The shared key is used by the encryption algorithm to generate a message signature for message authentication. Note that producer registration fails if the producer is set up with a shared key and you enter an incorrect shared key here. The shared key can contain between 10 and 20 alphanumeric characters. This key is also used when registering a producer using the Federated Portal Adapter (FPA). The Shared Key is also known as the HMAC key.

4. Click **OK**.

   The new producer appears in the connection table.

21.4.2 Registering an Oracle PDK-Java Producer Using WLST

Use the WLST command `registerPDKJavaProducer` to create a connection to a PDK-Java portlet producer and register the producer with your WebCenter Portal application. For command syntax and examples, see the "registerPDKJavaProducer" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

> **See Also:** `deregisterPDKJavaProducer`, `listPDKJavaProducers`, `refreshProducer`

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

21.4.3 Registering an Oracle PDK-Java Portlet Producer in WebCenter Portal

To register portlet producers in WebCenter Portal, you must have the following roles and permissions:

- **`AppConnectionManager` role**—Enables you to manage portlet producers.

  By default, users with the `Administrator` role in Portal Builder are assigned this role; and therefore, administrators can configure portlet producers. You can grant any other user this capability through Fusion Middleware Control or using the WLST command `grantAppRole`. For example, the following `grantAppRole` WLST command grants the `AppConnectionManager` role to the user `monty`:

  ```
  grantAppRole(appStripe='webcenter', appRoleName='AppConnectionManager',
  principalClass='weblogic.security.principal.WLSUserImpl',
  principalName='monty')
  ```

  See Section 32.6.2.1, "Granting Application Roles Using Fusion Middleware Control."

- **`Application - Manage Configuration` permission**—Enables access to the Portal Builder Administration page. See Chapter 49, "Managing Security Across Portals."

To register an Oracle PDK-Java portlet producer in WebCenter Portal:

1. Open WebCenter Portal Administration.

   For more information, see Section 47.2, "Accessing the Portal Builder Administration Page."

2. Click **Tools and Services**, and then select **Portlet Producers**.

   Alternatively, use the following URL, and then select Portlet Producers:

   `http://`*`host`*`:`*`port`*`/webcenter/portal/admin/tools`

3. On the menu bar, click **Register**.
4. In the Register Portlet Producer page, enter connection details for the Oracle PDK-Java portlet producer. For details, see Table 21–6.
5. Click **Test** to verify that the server details you provided are correct.

   If the server is contactable, a success message is displayed. If the server is down or the host information is incorrect or no longer valid, a connection failure message is displayed.

   > **Note:** The test performs a simple server (host/port) PING test. To verify whether the producer is accessible, access the producer's test page in your browser. For more information, see Section 21.5, "Testing Oracle PDK-Java Producer Connections."

6. Click **Ok**.

21.4.4 Registering an Oracle PDK-Java Portlet Producer in WebCenter Portal Framework Applications

For information about registering an Oracle PDK-Java portlet producer in Portal Framework applications at design time, see the "Registering an Oracle PDK-Java

Portlet Producer with a WebCenter Portal Framework Application" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

For information about registering an Oracle PDK-Java portlet producer in Portal Framework applications at runtime, see Section 21.9.1, "Registering Portlet Producers with the Administration Console."

21.5 Testing Oracle PDK-Java Producer Connections

To verify an Oracle PDK-Java producer connection, run the producer URL in a browser window in the following format:

`http://host_name:port_number/context-root/providers/producer_name`

For example:

```
http://domain.example.com:7778/axyz/providers/sample
```

21.6 Editing Producer Registration Details

You can update producer registration details at any time.

If a producer moves to a different location, then you must reconfigure any connections you have defined to this producer. You can use Fusion Middleware Control or WLST to edit the URL property:

- WDSL URL for a WSRP producer
- URL End Point for an Oracle PDK-Java producer

To retain all the portlet customizations and personalizations that users make while working with WebCenter Portal applications, you must also migrate producer customizations and personalizations to the producer's new location. Use the WLST commands `exportPortletClientMetadata` and `importPortletClientMetadata` to migrate portlet client metadata to a different location.

If you are migrating a Portal Framework application, see Section 44.1.3, "Exporting Portlet Client Metadata for Portal Framework Applications" and Section 44.1.4, "Importing Portlet Client Metadata for Portal Framework Applications".

For WebCenter Portal, see Section 41.6.10.1, "Backing Up (Exporting) Portlet Client Metadata" and Section 41.6.10.1, "Backing Up (Exporting) Portlet Client Metadata".

This section includes the following topics:

- Section 21.6.1, "Editing Producer Registration Details Using Fusion Middleware Control"
- Section 21.6.2, "Editing Producer Registration Details Using WLST"
- Section 21.6.3, "Editing Producer Registration Details in WebCenter Portal"
- Section 21.6.4, "Editing Producer Registration Details in WebCenter Portal Framework Applications"
- Section 21.6.5, "Migrating WSRP Producer Metadata to a New WSDL URL"

21.6.1 Editing Producer Registration Details Using Fusion Middleware Control

To update connection details for a portlet producer:

1. Log in to Fusion Middleware Control and navigate to the home page for the WebCenter Portal application. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For the WebCenter Portal application - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.
3. From the list of services on the WebCenter Portal Service Configuration page, select **Portlet Producers**.
4. In the **Manage Portlet Producer Connections** section, select the producer you want to modify, and click **Edit**.
5. In the **Edit Portlet Producer Connection** section, modify connection details, as required. For more information, see:
   - Table 21–1, " WSRP Producer Connection Parameters"
   - Table 21–6, " Oracle PDK-Java Producer Connection Parameters"
6. Click **OK**.

21.6.2 Editing Producer Registration Details Using WLST

Use the following WLST commands to edit portlet producer connections:

- **WSRP producers** - `setWSRPProducer`
- **PDK-Java producers** - `setPDKJavaProducer`

For command syntax and examples, see the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

21.6.3 Editing Producer Registration Details in WebCenter Portal

In WebCenter Portal, you can access and revise many of the registration details provided for a portlet producer.

To edit portlet producer registration details in WebCenter Portal:

1. Open WebCenter Portal Administration.

   For more information, see Section 47.2, "Accessing the Portal Builder Administration Page."
2. Click **Tools and Services**, and then select **Portlet Producers**.

   Alternatively, use the following URL, and then select Portlet Producers:

   `http://host:port/webcenter/portal/admin/tools`
3. Select the portlet producer that you want to edit.
4. On the menu bar, click **Edit**.

5. Edit the producer registration properties as required:
   - WSRP Producers
     - Table 21–1, " WSRP Producer Connection Parameters"
     - Table 21–5, " WSRP Portlet Producer Registration - Security"
   - Oracle PDK-Java Producers
     - Table 21–6, " Oracle PDK-Java Producer Connection Parameters"

   You cannot edit the **Producer Name** or **Producer Type**.

   Note: While it is possible to edit the value of the **WSDL URL** or **URL Endpoint**, for example, if the producer port has changed, you can point to a different producer only if the new producer has access to the persistence store of the old producer, or if the persistence store of the old producer has been migrated to that of the new producer. For more information, see Section 41.6.5, "Backing up and Restoring Other Schema Data (ACTIVITIES and PORTLET)".

6. When you have changed all the necessary settings, you can click **Test** to verify that the server details you provided are correct.

   If the server is contactable, a success message is displayed. If the server is down or the host information is incorrect or no longer valid, a connection failure message is displayed.

   Note: The test performs a simple server (host/port) PING test.

7. When you are done, click **Ok**.

21.6.4 Editing Producer Registration Details in WebCenter Portal Framework Applications

For information about editing portlet producer registration details in Portal Framework applications at design time, see the "How to Edit Portlet Producer Registration Settings" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

For information about editing portlet producer registration details in Portal Framework applications at runtime, see Section 21.9.2, "Editing Portlet Producer Registration Details with the Administration Console."

21.6.5 Migrating WSRP Producer Metadata to a New WSDL URL

If you want to move a WSRP producer to a new WSDL URL, you can use the `exportPortletClientMetadata`, `setWSRPProducer`, and `importPortletClientMetadata` WLST commands to migrate the existing producer metadata to the new location.

To migrate WSRP producer metadata to a new URL endpoint:

1. Export the producer metadata, using the WLST command `exportPortletClientMetadata`. For command syntax and examples, see the

"exportPortletClientMetadata" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

2. Change the producer's WSDL URL, using the WLST command `setWSRPProducer`. For command syntax and examples, see the "setWSRPProducer" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.
3. Import the producer metadata, using the WLST command `importPortletClientMetadata`. For command syntax and examples, see the "importPortletClientMetadata" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

21.7 Editing the Portlet Client Configuration

The `adf-config.xml` file contains configuration information for WebCenter Portal services. Portlet client configuration details are specified in the `adf-portlet-config` section of the file.

Example 21–1 shows the `adf-portlet-config` element of the `adf-config.xml` file.

Example 21–1 The adf-portlet-config section of adf-config.xml

```
<adf-portlet-config xmlns="http://xmlns.oracle.com/adf/portlet/config">
  <supportedLocales>
    <value>en</value>
    <value>fr</value>
    <value>de</value>
    <value>es</value>
  </supportedLocales>
  <portletTechnologies>

<value>oracle.portlet.client.containerimpl.web.WebPortletTechnologyConfig</value>
<value>oracle.portlet.client.containerimpl.wsrp.WSRPPortletTechnologyConfig</value
>
  </portletTechnologies>
  <defaultTimeout>20</defaultTimeout>
  <minimumTimeout>1</minimumTimeout>
  <maximumTimeout>300</maximumTimeout>
  <resourceProxyPath>/portletresource</resourceProxyPath>
  <cacheSettings>
    <maxSize>10000000</maxSize>
    <subscriber default="true">
      <systemLevel>
        <maxSize>5000000</maxSize>
      </systemLevel>
      <userLevel>
        <maxSize>8000000</maxSize>
      </userLevel>
    </subscriber>
  </cacheSettings>
</adf-portlet-config>
```

Application developers can edit the `adf-config.xml` file for an application and edit the portlet client configuration. However, this requires that the application be redeployed after the changes are made. To edit the configuration of the portlet client at runtime, without having to redeploy the application, you can use WLST commands.

Use the WLST command `setPortletClientConfig` to edit the portlet client configuration information. For command syntax and examples, see the "setPortletClientConfig" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

After using this WLST commands, you must restart the Managed Server on which the WebCenter application is deployed. For details, see Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

> **See Also:** `listPortletClientConfig`, `getPortletClientConfig`

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

21.8 Deregistering Producers

You can deregister producers at any time but, before doing so, consider any impact to the WebCenter Portal application as portlets associated with a deregistered producer no longer work. Check the *Portlets Producer Invocation* metric to see how frequently the producer is being used. For more information, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

When you deregister a producer, registration data is removed from both the WebCenter Portal application and the remote producer:

- WebCenter Portal application - The producer connection is deleted and producer metadata is also deleted.
- Remote producer - Portlet instances are deleted (not the portlets themselves).

Portlet instances are not removed from WebCenter Portal application pages. In place of the portlet, users see a "`Portlet unavailable`" message.

> **Note:** Consider deleting the external application associated with this portlet producer *if* the application's sole purpose was to support this producer. See Section 23.6, "Deleting External Application Connections."

This section includes the following topics:

- Section 21.8.1, "Deregistering Producers Using Fusion Middleware Control"
- Section 21.8.2, "Deregister Producers Using WLST"
- Section 21.8.3, "Deregistering Producers in WebCenter Portal"
- Section 21.8.4, "Deregistering Producers in WebCenter Portal Framework Applications"

21.8.1 Deregistering Producers Using Fusion Middleware Control

To deregister a portlet producer:

1. Log in to Fusion Middleware Control and navigate to the home page for the WebCenter Portal application. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"

- Section 6.3, "Navigating to the Home Page for Portal Framework Applications"

2. Do one of the following:
   - For the WebCenter Portal application - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.
3. From the list of services on the WebCenter Portal Service Configuration page, select **Portlet Producers**.
4. Select the name of the producer you want to remove, and click **Delete**.

The connection details are removed. Portlets associated with this producer are no longer accessible within the WebCenter Portal application.

21.8.2 Deregister Producers Using WLST

Use the following WLST commands to deregister portlet producer connections:

- **WSRP producers** - `deregisterWSRPProducer`
- **PDK-Java producers** - `deregisterPDKJavaProducer`

Use the following WLST commands to deregister the out-of-the-box or sample producers provided with Oracle WebCenter Portal:

- **Out-of-the-box producers** - `deregisterOOTBProducers`
- **Sample producers** - `deregisterSampleProducers`

For command syntax and examples, see the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

21.8.3 Deregistering Producers in WebCenter Portal

If you no longer want to use a particular producer in WebCenter Portal, you can deregister the producer. When you deregister a producer, registration data is removed from both WebCenter Portal and the remote producer:

- WebCenter Portal— The producer connection is deleted and producer metadata is also deleted.
- Remote producer—Portlet instances are deleted (not the portlets themselves).

To deregister a portlet producer:

1. Open WebCenter Portal Administration.

   For more information, see Section 47.2, "Accessing the Portal Builder Administration Page."
2. Click **Tools and Services**, and then select **Portlet Producers**.

   Alternatively, use the following URL, and then select Portlet Producers:

   `http://host:port/webcenter/portal/admin/tools`
3. Select the portlet producer that you want to remove.

4. From the menu bar, click **Deregister**.
5. In the Delete Confirmation dialog, click **Deregister** to complete the deregistration process.

21.8.4 Deregistering Producers in WebCenter Portal Framework Applications

For information about deregistering portlet producers in Portal Framework applications at design time, see the "How to Delete a Portlet Producer" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

For information about deregistering portlet producers in Portal Framework applications at runtime, see Section 21.9.3, "Deregistering Portlet Producers with Administration Console."

21.9 Managing Portlet Producers with the Administration Console

To be able to register and manage portlet producers in a Portal Framework application, a user must be assigned the `AppConnectionManager` role. By default, users with the `Administrator` role have the `AppConnectionManager` role; and therefore, application administrators can configure portlet producers through the administration console. See also, Section 43.4.4, "Managing Application Roles and Permissions."

When you register a portlet producer, all the portlets owned by that producer automatically become available through the application's resource catalog. Once registered, users with appropriate edit page privileges, are then able to add the producer's portlets to their pages. Users who want access to a particular portlet but cannot find it in the resource catalog must ask an administrator to register the associated producer.

This section includes the following:

- Section 21.9.1, "Registering Portlet Producers with the Administration Console"
- Section 21.9.2, "Editing Portlet Producer Registration Details with the Administration Console"
- Section 21.9.3, "Deregistering Portlet Producers with Administration Console"

Note: System administrators can also register portlet producers for Portal Framework application, using Fusion Middleware Control and WLST commands. For details, see Section 21.2, "Registering WSRP Producers."

21.9.1 Registering Portlet Producers with the Administration Console

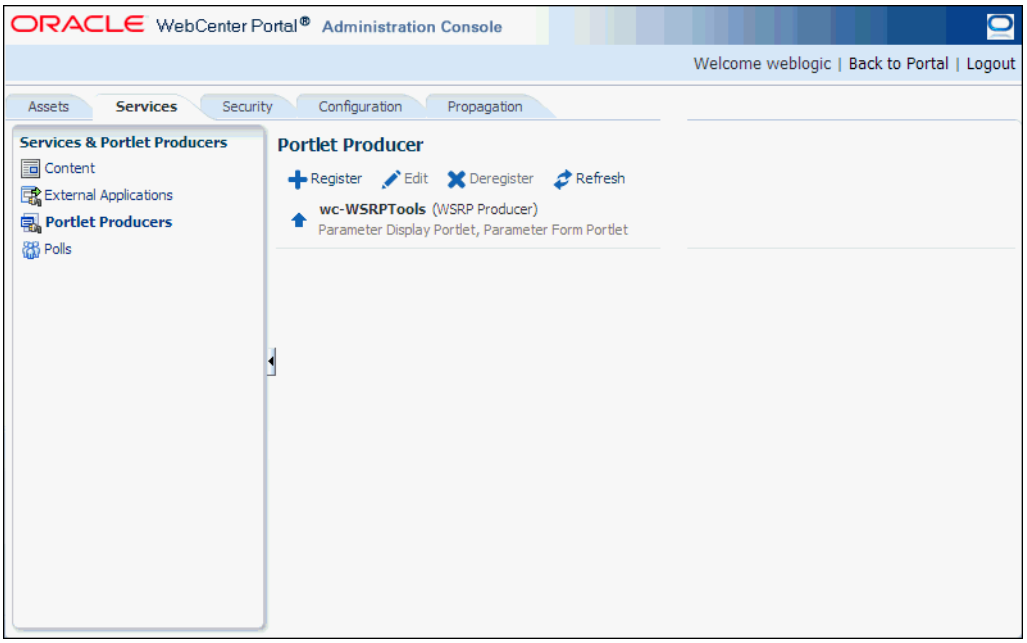
To register a portlet producer at runtime for a Portal Framework application:

1. Open the **Services** tab in administration console.

   See also, Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. In the navigation panel on the left, click **Portlet Producers** (Figure 21–1).

Figure 21–1 Administration Console - Portlet Producers



3. On the menu bar, click **Register**.
4. Enter connection details for the portlet producer.

   If you need help with one or more fields, refer to the following tables:

   WSRP Producers

   - Table 21–1, " WSRP Producer Connection Parameters"
   - Table 21–5, " WSRP Portlet Producer Registration - Security"

   Oracle PDK Java Producers

   - Table 21–6, " Oracle PDK-Java Producer Connection Parameters"
5. Click **Test** to verify your connection details.
6. Click **OK** to register the portlet producer.

21.9.2 Editing Portlet Producer Registration Details with the Administration Console

To modify a portlet producer at runtime for a Portal Framework application:

1. Open the **Services** tab in administration console.

   See also, Section 43.2, "Accessing the Administration Console for Portal Framework Applications."
2. In the navigation panel on the left, click **Portlet Producers** (Figure 21–1).
3. Select the portlet producer required, and then in the menu bar click **Edit** to update its connection details.
4. Edit the producer registration properties as required:
   - WSRP Producers
     - Table 21–1, " WSRP Producer Connection Parameters"
     - Table 21–5, " WSRP Portlet Producer Registration - Security"

- Oracle PDK-Java Producers
  - Table 21–6, " Oracle PDK-Java Producer Connection Parameters"

You cannot edit the **Producer Name** or **Producer Type**.

5. Click **Test** to verify the amended connection details.
6. Click **OK** to save your changes.

21.9.3 Deregistering Portlet Producers with Administration Console

To delete a portlet producer at runtime for a Portal Framework application:

1. Open the **Services** tab in administration console.

   See also, Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. In the navigation panel on the left, click **Portlet Producers** (Figure 21–1).
3. Select the portlet producer required, and then in the menu bar click **Deregister** to deregister the producer.

   This removes registration data from both the Portal Framework application and the remote producer.

   Deregistering *does not* remove portlet instances from Portal Framework application pages. Instead of the portlet, users see a "`Portlet unavailable`" message.

   You should also consider deleting the external application associated with this portlet producer if the application's sole purpose is to support this producer. See Section 23.4.2, "Editing and Deleting External Applications."

4. In the Delete Confirmation dialog, click **Deregister** to complete the deregistration process.

21.10 Working with the Producer Registration Task Flow

To manage portlet producers through the Producer Registration task flow, you must have the following roles and permissions:

- **`AppConnectionManager` role**—Enables you to manage portlet producers.
- **`View Page` permission**—Enables access to the page containing the task flow.

This section includes the following topics:

- Section 21.10.1, "Adding the Producer Registration Task Flow to a Page"
- Section 21.10.2, "Registering a Portlet Producer Using the Producer Registration Task Flow"
- Section 21.10.3, "Setting Producer Registration Task Flow Properties"

21.10.1 Adding the Producer Registration Task Flow to a Page

To add a Producer Registration task flow:

1. Go to the page where you want to add the task flow, and open it in Composer.

   For more information, see the "Opening a Page in the Page Editor (Composer)" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. In Composer, click the **Add Content** button.

3. In the Add Content dialog, open **UI Components** and then **Portlets**.
4. In the Portlets folder, open **Portlet Administration**.
5. In the Portlet Administration folder, click the **Add** link next to the **Producer Registration** task flow. This adds the task flow to the page.
6. Click **Close** to exit Add Content dialog.

21.10.2 Registering a Portlet Producer Using the Producer Registration Task Flow

When you have added the Producer Registration task flow to your page, you can register any portlet producer as described in the following tables:

- **WSRP Producers**
  - Table 21–1, " WSRP Producer Connection Parameters"
  - Table 21–5, " WSRP Portlet Producer Registration - Security"
- **Oracle PDK-Java Producers**
  - Table 21–6, " Oracle PDK-Java Producer Connection Parameters"

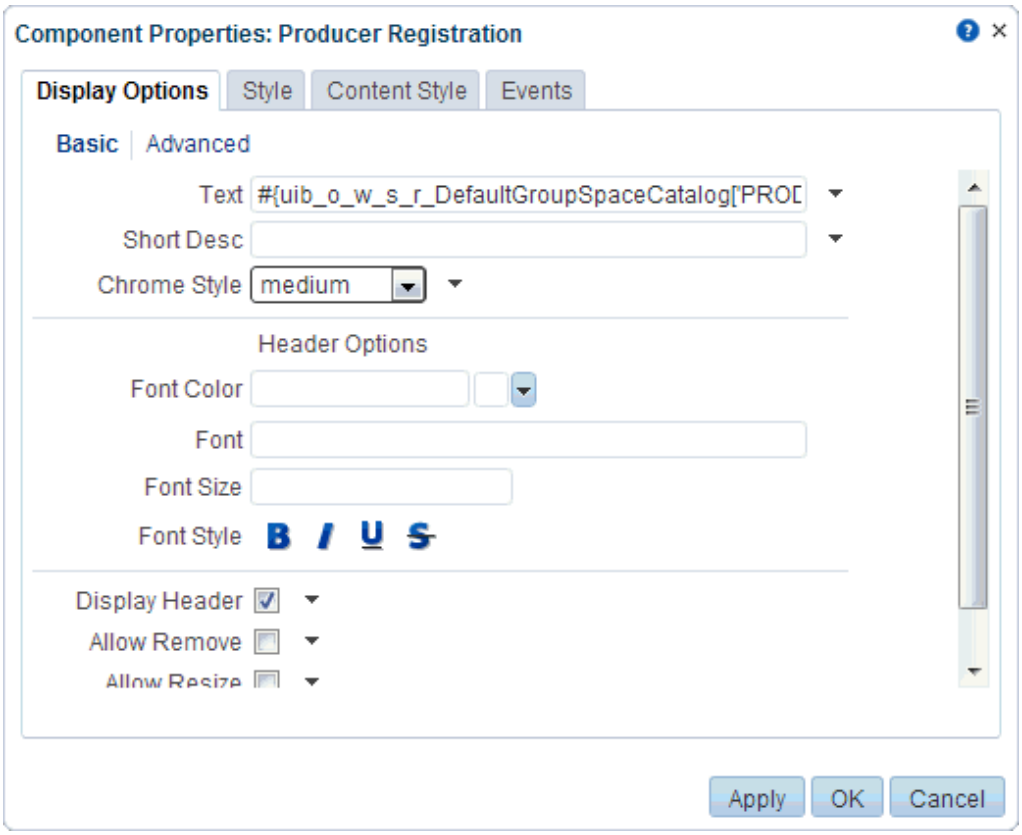
See also Section 21.6.3, "Editing Producer Registration Details in WebCenter Portal" and Section 21.8.3, "Deregistering Producers in WebCenter Portal."

21.10.3 Setting Producer Registration Task Flow Properties

The Producer Registration task flow has associated properties, which users with sufficient privileges can access through the Component Properties dialog in Composer (Figure 21–2).

For information about task flow properties and how to access them, see the "Modifying Components" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Figure 21–2 Producer Registration Task Flow Component Properties



21.11 Deploying Portlet Producer Applications

To deploy a portlet producer to an Oracle WebLogic Managed Server instance, you can use Fusion Middleware Control, Oracle WebLogic Server Administration Console, or WLST.

This section includes the following topics:

- Section 21.11.1, "Understanding Portlet Producer Application Deployment"
- Section 21.11.2, "Preparing Portlet Producer Applications for Deployment"
- Section 21.11.3, "Deploying Portlet Producer Applications Using Fusion Middleware Control"
- Section 21.11.4, "Deploying Portlet Producer Applications Using Oracle WebLogic Server Administration Console"
- Section 21.11.5, "Deploying Portlet Producer Applications Using WLST"
- Section 21.11.6, "Deploying Portlet Producer Applications Using Oracle JDeveloper"

For more information about deploying applications, see the "Deploying Applications" chapter in the *Oracle Fusion Middleware Administrator's Guide*.

21.11.1 Understanding Portlet Producer Application Deployment

You can deploy your Portlet Producer application to any Oracle WebLogic Managed Server instance that is configured to support WebCenter Portal portlet producers. To

deploy an application to a managed server, you can use Oracle Enterprise Manager Fusion Middleware Control, Oracle WebLogic Administration Console, or WLST. For more information about these administration tools, see Section 1.13, "Oracle WebCenter Portal Administration Tools."

21.11.2 Preparing Portlet Producer Applications for Deployment

WebCenter Portal provides a predeployment tool that adds the required configuration to a portlet producer application's EAR file to expose the portlets over WSRP. The predeployment tool must be run in the following circumstances:

- You created the application's WAR file outside of JDeveloper.
- You created the application's WAR file in JDeveloper, but selected to not expose the application as a WSRP application. That is, you selected **No** in the Select deployment type dialog.

To add the required configuration to a portlet producer application's EAR file to expose the portlets over WSRP, run the WSRP producer predeployment tool located in the Middleware directory at *WCP_ORACLE_HOME*/webcenter/modules/oracle.portlet.server_11.1.1, as follows:

```
java -jar wsrp-predeploy.jar source EAR  target EAR
```

For JSR 286 portlets developed with servlet version 2.3, you must specify web proxies using the following command:

```
java -Dhttp.proxyHost=proxy host -Dhttp.proxyPort=proxy port -jar
wsrp-predeploy.jar source EAR target EAR
```

where:

- *proxy host* is the server to which your producer has been deployed.
- *proxy port* is the HTTP Listener port.
- wsrp-predeploy.jar is located in the *WCP_ORACLE_HOME*/webcenter/modules/oracle.portlet.server_11.1.1 directory.
- *source EAR* is the name of the JSR 286 EAR file.
- *target EAR* file is the name of the new EAR file to be created. If the file name for the targeted EAR file is not specified, then a new EAR file called WSRP-*source EAR* is produced.

In the following example a web proxy is specified:

```
java -Dhttp.proxyHost=myhttpproxy.com -Dhttp.proxyPort=80 -jar wsrp-predeploy.jar
wsrp-samples.ear
```

This example produces WSRP-wsrp-samples.ear.

The wsrp-predeploy.jar predeployment tool makes all the necessary changes to a JSR 286 portlet to be able to deploy it to the Oracle portlet container and expose it as a WSRP producer. Here are some examples of what the predeployment tool does:

- Creates the wsdldeploy directory in the java.io.tmpdir folder.
  - On UNIX, the default value of this property is /tmp or /var/tmp
  - On Microsoft Windows, the default value of this property is c:\temp.

- Unpacks the EAR file into `wsdldeploy/EAR`.
- Unpacks the WAR files into `wsdldeploy/[warfilename.war]/`.
- Inserts `WEB-INF/WSDLs` into the unpacked application.
- Modifies `WEB-INF/web.xml` in the unpackaged WAR files.
- Inserts or modifies `WEB-INF/webservices.xml` in the WAR files.
- Inserts or modifies `WEB-INF/oracle-webservices.xml` in the WAR files.
- Repackages the WARs and builds a new EAR file.

21.11.3 Deploying Portlet Producer Applications Using Fusion Middleware Control

For information about deploying a portlet producer application using Fusion Middleware Control, see Section 42.1.6.4, "Deploying Applications Using Fusion Middleware Control."

21.11.4 Deploying Portlet Producer Applications Using Oracle WebLogic Server Administration Console

For information about deploying a portlet producer application using Oracle WebLogic Server Administration Console, see Section 42.1.6.6, "Deploying Applications Using the WLS Administration Console."

21.11.5 Deploying Portlet Producer Applications Using WLST

For information on deploying a portlet application using the WLST command, see Section 42.1.6.5, "Deploying Applications Using WLST."

21.11.6 Deploying Portlet Producer Applications Using Oracle JDeveloper

You can deploy portlet applications to an Oracle WebLogic Managed Server instance directly from the development environment using Oracle JDeveloper, if you have the necessary credentials to access the WebLogic server. For more information, see the "Deploying Portlet Producers" chapter in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

21.12 Configuring WebCenter Services Portlets

WebCenter Portal provides social networking and personal productivity features through tools and services, which, in turn, expose subsets of their features and functionality through task flows. These task flows are readily available for use in WebCenter Portal and other Portal Framework applications. However, application developers using other products, such as Oracle Portal, Oracle WebLogic Portal, and Oracle WebCenter Interaction, may also want to expose these same features within those applications.

WebCenter Services Portlets is a preconfigured, out-of-the-box producer that enables you to expose WebCenter Portal tools and services task flows to other applications as WSRP portlets or pagelets.

The following task flows are provided as portlets by WebCenter Services Portlets:

- Document Manager—Displays folders, files, and wikis from the WebCenter Content repository

- Blogs—Displays blog posts from a selected location in the WebCenter Content repository
- Discussion Forums—Displays all discussions and their respective replies and enables users to perform various operations based on their privileges
- Announcements—Displays all current announcements and enables users to perform various operations based on their privileges
- Lists—Displays user-created lists and provides controls for creating lists and adding list data
- Polls Manager—Enables users to perform administrative operations on polls
- Take Polls—Displays the most recently published available poll, or a specific poll identified by the pollId parameter
- Mail—Displays a mail inbox
- Activity Stream—Provides an overview of the most recent activities performed by a user's connections
- Tag Cloud—Displays a tag cloud, which is a visual depiction of all the tags used on the page

WebCenter Services Portlets starts life as a Portal Framework application. This application includes several pages, one for each of the exposed task flows. The application is then portletized, using the Oracle JSF Portlet Bridge, and deployed to the `WC_Portlet` managed server.

After installation of WebCenter Portal, WebCenter Services Portlets is automatically available for use. However, for the portlets and pagelets to work correctly there are some configuration steps that must be completed.

This section includes the following topics:

- Section 21.12.1, "Configuring Back-End Connections"
- Section 21.12.2, "Configuring Security for WebCenter Services Portlets"
- Section 21.12.3, "Troubleshooting WebCenter Services Portlets"

21.12.1 Configuring Back-End Connections

Most of the tools and services included in WebCenter Services Portlets require connections to back-end servers to be fully functional. For example, documents requires a connection to an Oracle WebCenter Content repository and discussions and announcements require a connection to a discussions server for Oracle WebCenter Portal. These connections must be configured before application developers can start to consume the portlets and pagelets provided by the producer.

This section includes the following topics:

- Section 21.12.1.1, "Configuring the Documents Content Repository Connection"
- Section 21.12.1.2, "Configuring the Discussions and Announcements Connection"
- Section 21.12.1.3, "Configuring the Mail Connection"

21.12.1.1 Configuring the Documents Content Repository Connection

WebCenter Services Portlets includes portlets for task flows provided by documents, blogs, and wikis. These portlets require a connection to a back-end WebCenter Content repository.

For general information about configuring WebCenter Content, see Chapter 9, "Managing Content Repositories."

There are two ways to create a connection to a WebCenter Content repository:

- Using Fusion Middleware Control. For more information, see Section 9.6.2, "Registering Content Repositories Using Fusion Middleware Control."
- Using the WLST command line tool. For more information, see Section 9.6.3, "Registering Content Repositories Using WLST."

> **Note:** WebCenter Services Portlets uses the WebCenter Content repository identified as the active or default connection, so you must ensure that you have set the appropriate connection as the default. For more information, see Section 9.7, "Changing the Active (or Default) Content Repository Connection."

21.12.1.2 Configuring the Discussions and Announcements Connection

WebCenter Services Portlets includes portlets for task flows provided by the discussions and announcements. These task flows require a connection to a back-end discussions server for Oracle WebCenter Portal. Discussions and announcements use the same connection. Oracle WebCenter Portal's discussion server is installed automatically with Oracle Fusion Middleware, but you must create the connection to the server.

For general information about configuring discussions and announcements, see Chapter 12, "Managing Announcements and Discussions."

There are two ways to create a connection to an Oracle WebCenter Portal Discussions server:

- Using Fusion Middleware Control. For more information, see Section 12.3.1, "Registering Discussions Servers Using Fusion Middleware Control."
- Using the WLST command line tool. For more information, see Section 12.3.2, "Registering Discussions Servers Using WLST."

> **Note:** WebCenter Services Portlets uses the discussions server identified as the active or default connection, so you must ensure that you have set the appropriate connection as the default. For more information, see Section 12.4, "Choosing the Active Connection for Discussions and Announcements."

21.12.1.3 Configuring the Mail Connection

WebCenter Services Portlets includes portlets for the Mail task flow. Mail requires a connection to a back-end mail server. This server can be the Microsoft Exchange Server or any mail server that supports IMAP4 and SMTP.

For general information about configuring the mail, see Chapter 15, "Managing Mail."

There are two ways to create a connection to your mail server:

- Using Fusion Middleware Control. For more information, see Section 15.4.1, "Registering Mail Servers Using Fusion Middleware Control."
- Using the WLST command line tool. For more information, see Section 15.4.2, "Registering Mail Servers Using WLST."

Note: WebCenter Services Portlets uses the mail server identified as the active or default connection, so you must ensure that you have set the appropriate connection as the default. For more information, see Section 15.5, "Choosing the Active (or Default) Mail Server Connection."

21.12.2 Configuring Security for WebCenter Services Portlets

WebCenter Services Portlets should be secured to ensure that users cannot access information that they do not have privileges to access. As a WSRP producer, WebCenter Services Portlets uses WS-Security to ensure identity propagation.

For information about how to configure WS-Security for a WSRP portlet producer, see Section 37.1, "Securing a WSRP Producer."

Note: After attaching the required security policy, you must restart the `WC_Portlet` managed server.

21.12.3 Troubleshooting WebCenter Services Portlets

This section includes the following topics:

- Section 21.12.3.1, "Rich Text Editor Not Working for Document Manager and Blogs Portlets"
- Section 21.12.3.2, "Cannot Manage Lists in the Lists Portlet"
- Section 21.12.3.3, "Portlet Uses Incorrect Time Zone or Date and Time Format"
- Section 21.12.3.4, "Portlet Displays Remote Portlet Communication Error"

21.12.3.1 Rich Text Editor Not Working for Document Manager and Blogs Portlets

Blogs and wikis use a rich text editor (CKEditor) that requires access to various resources at runtime.

If the WebCenter Services Portlets producer is behind a firewall that the end-user browser cannot get through, the URLs for the resources required by CKEditor cannot be accessed and the editor will not display in the Blogs portlet or in the Document Manager portlet when viewing wikis.

To resolve this issue:

1. Create an application that includes the CKEditor resources, for example by using the WebCenter Portal Framework Application template.
2. Deploy the application to the same server that is used by the application that is consuming WebCenter Services Portlets.
3. Run the WebCenter Services Portlets producer application.
4. In the administration console, edit the **Base Resource URL** to point to the new application.

   For more information about how to do this, see Section 43.3.5, "Choosing the Default Base Resource URL."
5. Refresh the browser displaying the consumer application. The rich text editor should now display correctly as the CKEditor resources are available on the same side of the firewall as the consumer application.

21.12.3.2 Cannot Manage Lists in the Lists Portlet

Out of the box, only users who are members of the `Administrator` role can create, edit, and delete lists and edit list data. If a user is not a member of the `Administrator` role, he or she will only be able to view lists.

To enable a user who is not a member of the `Administrator` role to manage lists, use Fusion Middleware Control to assign the user the `manage` permission on the `ListPermission` role:

```
<permission>
  <class>oracle.webcenter.list.model.security.ListPermission</class>
  <name>/oracle/webcenter/list/templates/lists/.*</name>
  <actions>manage</actions>
</permission>
```

1. In Fusion Middleware Control, under **Application Deployments**, right-click the **services-producer** application and choose **Security** and then **Application Policies** from the context menu.
2. Click **Create**.
3. In the Grantee section, click **Add**.
4. In the Add Principal dialog, locate the user or role to which you want to add the list permission.
5. Click **OK**.
6. In the Permissions section, click **Add**.
7. In the Add Permission dialog, from the **Permission Class** drop-down list, select **oracle.webcenter.list.model.security.ListPermission**.
8. Click the **Search system security grants** icon.
9. In the Search Results section, select the **Resource Name** with the `manage` **Permission Action**.
10. Click **Continue**.
11. In the confirmation dialog, click **Select** to confirm the permission for the target user or role.
12. Click **OK**.

21.12.3.3 Portlet Uses Incorrect Time Zone or Date and Time Format

Currently, user preferences are not propagated from the portlet consumer to the WebCenter Services Portlets producer. This means that WebCenter Services Portlets always use the time zone and date and time format preferences set on the server where the producer is deployed, regardless of the settings users specify in the consumer application.

21.12.3.4 Portlet Displays Remote Portlet Communication Error

If a portlet provided by WebCenter Services Portlets displays a Remote Portlet Communication Error, this is typically because the WS-Security was not configured on the producer.

For information about how to resolve this problem, see Section 21.12.2, "Configuring Security for WebCenter Services Portlets."

21.13 Troubleshooting Portlet Producer Issues

This section includes the following sub sections:

- Section 21.13.1, "Producer Registration Fails for a WebCenter Portal Framework Application"
- Section 21.13.2, "Portlet Unavailable: WSM-00101 Exception"

21.13.1 Producer Registration Fails for a WebCenter Portal Framework Application

This section describes producer registration and portlet unavailability issues.

Problem

You are unable to register a WSRP producer.

Solution

Ensure the following:

- Back-end producer is up and running. To test the producer, access the WSDL URL of the producer through a browser window. See, Section 21.3, "Testing WSRP Producer Connections."
- Producer application is packaged accurately. If not, then register the producer at design time (in JDeveloper), as described in the "Registering Portlet Producers with a WebCenter Portal Framework Application" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*, and redeploy the application, as described in Section 42.1, "Deploying Portal Framework Applications." After redeployment, verify that the packaged application includes the MBean, `ProducerManager`:

  1. In Fusion Middleware Control, from the **Application Deployment** menu, select **System MBean Browser**.
  2. In the Navigator, expand **Application Defined MBeans** > **oracle.webcenter.portlet** > **Application**: ***application_name*** > **Producer Manager > Producer Manager**.

- `PortletServletContextListener` is added to the `web.xml` file.

  For applications that support post deployment registration of producers, the producer must be registered at least once at design time. This adds `PortletServletContextListener` to the `web.xml` file, which registers the appropriate runtime MBeans to enable post deployment registration of producers. For example, see the text in **bold** in the following `web.xml` snippet:

```
<listener>
   <description>
      WebCenter Portlet Context Listener
   </description>
   <display-name>
      WebCenterPortletContextListener
   </display-name>
   <listener-class>
      oracle.webcenter.portlet.listener.PortletServletContextListener
   </listener-class>
</listener>
```

21.13.2 Portlet Unavailable: WSM-00101 Exception

Setting up the **User Name with Password** token profile in a WSRP portlet producer throws the exception `WSM-00101`.

Problem

If you configure the **User Name with Password Token** profile for a WSRP producer through Fusion Middleware Control (or WLST) while portlets associated with this producer are in use, the portlets display the following exception in the WebCenter Portal application:

```
oracle.wsm.common.sdk.WSMException: WSM-00101:
The specified Keystore file
/keys/user_projects/domains/pv_0309/config/fmwconfig/default-keystore.jks
cannot be found; it either does not exist or its path is not included in the
application classpath.
```

Solution

Ensure that you have configured the default keystore in your portlet producer. For information, see Section 37.1.3, "Setting Up the Keystores."

22

Managing the Pagelet Producer

Oracle WebCenter Portal's Pagelet Producer (previously called Oracle WebCenter Ensemble) provides a collection of useful tools that facilitate dynamic pagelet development and deployment. The Pagelet Producer proxy provides users with external access to internal resources including internal applications and secured content. Using the Pagelet Producer, you can expose WSRP and Oracle JPDK portlets and OpenSocial gadgets as pagelets for use in any web page or application.

This chapter describes how to register, edit and deploy pagelets using the Pagelet Producer Administrative Console.

This chapter includes the following sections:

- Section 22.1, "About Pagelet Producer"
- Section 22.2, "Registering Pagelet Producer"
- Section 22.3, "Registering WSRP and Oracle JPDK Portlet Producers in the Pagelet Producer"
- Section 22.4, "Configuring the Trust Service Identity Asserter"
- Section 22.5, "Managing Import, Export, Backup and Recovery of Pagelet Producer Components"

For information about developing and deploying pagelets, see the "Creating Pagelets with Pagelet Producer" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console and the `Administrator` role in the deployed application:

- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.
- **Portal Framework application**: `Administrator` role granted through the Administration Console.

For more information about roles and permissions, see Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

22.1 About Pagelet Producer

This section is an introduction to Pagelet Producer concepts and features and includes the following topics:

22.1.1 Overview

Oracle WebCenter Portal's Pagelet Producer (previously known as Oracle WebCenter Ensemble) can be used to create *pagelets* to expose platform-specific portlets in other Web environments, including WebCenter Portal and Framework applications. Pagelet Producer provides a collection of useful tools and features that facilitate dynamic pagelet development. For information about Pagelet Producer architecture, component descriptions, and Pagelet Producer requirements , see the "About Pagelet Producer" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

Pagelet Producer registration is dynamic. Additions and updates to existing producers are immediately available; in most cases, it is not necessary to restart the application (WebCenter Portal or your own Portal Framework application) or the managed server.

Note: In the current release, only a single administrator can modify Pagelet Producer administrative settings at any given time. Concurrent edits will result in only one edit succeeding. However, data integrity will always be preserved.

22.1.2 Using the Pagelet Producer Console

The **Pagelet Producer Console** is a browser-based administration tool used to create and manage the various objects in your Pagelet Producer deployment. From the Console you can register Web applications as resources, create pagelets, manage proxy and transformation settings, and more.

- From WebCenter Portal, you can access the Pagelet Producer Console from the Shared Assets tab.

  Note: Pagelet Producer Console supports the standard administration languages and Dutch only. If you configure the browser language to something other than one of these languages, it will revert to the language defined for the current server.

- The Pagelet Producer Console is also accessible from any Web browser at the following URL:

  `http://<host>:<port>/pagelets/admin`

  The Pagelet Producer Console can also be launched in accessibility mode at:

  `http://<host>:<port>/pagelets/admin/accessible`

For more information about using the Pagelet Producer Console to configure Pagelet Producer, see the "Configuring Pagelet Producer Settings" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

22.1.3 Exposing WSRP and Oracle JPDK Portlets

Using the Pagelet Producer, you can expose WSRP and Oracle JPDK portlets as pagelets for use in any web page or application.

After setting up the Pagelet Producer as described in Section 22.2, "Registering Pagelet Producer," follow the steps below to import WSRP or Oracle JPDK portlets:

1. Register the portlet producer with the Pagelet Producer as described in Section 22.3, "Registering WSRP and Oracle JPDK Portlet Producers in the Pagelet Producer."
2. This automatically creates a resource and pagelets in the Pagelet Producer Console based on the portlet definitions for the producer. For details on resource settings, see the "Creating Pagelets with Pagelet Producer" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.
3. To modify the imported resource or the associated pagelets, you must make a copy of the imported resource; for details, see Section 22.3.1, "Using WSRP and Oracle JPDK Portlets."

22.1.4 Exposing OpenSocial Gadgets

Using the Pagelet Producer, you can expose OpenSocial gadgets as pagelets for use in any web page or application. For details, see the "Working with OpenSocial Gadgets" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

22.1.5 Exposing Oracle WebCenter Interaction Portlets

The Pagelet Producer can be used as a portlet provider for Oracle WebCenter Interaction. There are several configuration pages that allow you to define CSP settings for use with Oracle WebCenter Interaction. For details on configuring these settings and objects, see the "Creating Pagelets with Pagelet Producer" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

1. Configure the Pagelet Producer settings for use with the Oracle WebCenter Interaction Credential Mapper, SOAP API service and image service on the CSP Settings page in the Pagelet Producer Console.
2. Set up the Pagelet Producer's connection to the server hosting the portlet code by creating a "CSP" resource.
3. Create pagelets for the Oracle WebCenter Interaction portlets.

22.2 Registering Pagelet Producer

This section describes how to register and configure Pagelet Producer using Fusion Middleware Control and WLST commands. This section includes the following subsections:

- Section 22.2.1, "Registering Pagelet Producer Using Fusion Middleware Control"
- Section 22.2.2, "Registering Pagelet Producer Using WLST"
- Section 22.2.3, "Configuring the Pagelet Producer Service for WebCenter Portal"
- Section 22.2.4, "Registering Pagelet Producer Using WebCenter Portal"
- Section 22.2.5, "Redeploying Pagelet Producer to a Different Context"

For information about developing and deploying pagelets, see the "Creating Pagelets with Pagelet Producer" section in the *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

22.2.1 Registering Pagelet Producer Using Fusion Middleware Control

To register Pagelet Producer using Fusion Middleware Control:

1. Log in to Fusion Middleware Control and navigate to the home page WebCenter Portal or your Portal Framework application. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Register Producer**.
   - For Framework applications - From the **Application Deployment** menu, select **WebCenter Portal**, then **Register Producer**.
3. Enter connection details for the Pagelet Producer (Table 22–1).

Table 22–1 Pagelet Producer Connection Parameters

Field	Description
Connection Name	A unique name to identify this Pagelet Producer instance within the application. The name must be unique across all WebCenter Portal connection types. The name specified here appears in Composer under the Mash-ups > Pagelet Producers folder (by default).
Producer Type	Select **Pagelet Producer**.

Server URL	The URL to Pagelet Producer. The URL must include a fully-qualified domain name. Use the following syntax:
	`<protocol>://<host_name>:<port_number>/pagelets/`
	For example:
	`http://myhost.com:7778/pagelets/`
	If pagelets contain secure data, the registered URL must use the `https` protocol. For example:
	`https://myhost.com:7779/pagelets/`
	The context root can be changed from `/pagelets/` if necessary; for details, see Section 22.2.5, "Redeploying Pagelet Producer to a Different Context."
	Note: In WebCenter Portal, if the Pagelet Producer URL is protected by OAM, the URL to the pagelet catalog must be excluded (mapped directly without access control), or the catalog will appear to be empty when using REST. The pagelet catalog URL is `http://<host_name>:<port_number>/pagelets/api/v2/ensemble/pagelets`

4. Click **OK**. The new producer appears in the connection table.

22.2.2 Registering Pagelet Producer Using WLST

Use the `registerPageletProducer` command to register a Pagelet Producer for your portal or Framework application. For command syntax and examples, see the section

"registerPageletProducer" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

You can also use WLST to list or edit the current connection details.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

22.2.3 Configuring the Pagelet Producer Service for WebCenter Portal

This section describes how to set up the Pagelet Producer for use as a service by Oracle WebCenter Portal using the Oracle Configuration Wizard.

To set up the Pagelet Producer as a WebCenter Portal service:

1. Launch the **Configuration Wizard** by selecting **Oracle Fusion Middleware** > **Oracle WebLogic Server** > **Tools** > **Configuration Wizard**.
2. Select **Create a new WebLogic Domain** and then click **Next**.
3. Select **Base this domain on an existing template** and select the **Pagelet Producer domain template**. Confirm that the template location is correct and click **Next**.
4. Complete the domain configuration wizard. For details, see the online help.

All post deployment connection configuration is stored in the Oracle Metadata Services (MDS) repository. For more information, see Section 1.3.5, "Oracle WebCenter Portal Configuration Considerations." For detailed information about MDS, see the chapter "Managing the MDS Repository" in the *Oracle Fusion Middleware Administrator's Guide*.

Pagelet Producer stores all configuration data on a separate partition in the MDS schema of RCU. Typically, this schema is installed as part of the Oracle WebCenter Portal installation. This configuration data does not conflict with data that belongs to other services. When the Pagelet Producer domain template is deployed, the wizard prompts for connectivity information to the database in which the schema has been created. The names that the Pagelet Producer expects are:

- Datasource Name: `mds-PageletProducerDS`
- JNDI name: `jdbc/mds/PageletProducerDS`
- MDS partition name: `pageletproducer`

To use OpenSocial gadgets in conjunction with WebCenter Portal profile and activities features, you must manually configure the WebCenterDS data source to target the WC_Portlet server.

1. In the Oracle WebLogic Server Console, go to **Services** > **Data Source**.
2. Click on the **WebCenterDS** data source.
3. Go to the **Targets** tab.
4. Select the **WC_Portlet** server and click **Save**.

22.2.4 Registering Pagelet Producer Using WebCenter Portal

This section explains how to register Pagelet Producer in WebCenter Portal.

1. Log in to WebCenter Portal and click **Administration**.
2. Navigate to the **Configuration** tab and click **Services**.
3. On the Services and Providers page, click **Portlet Producers**.

4. Click **Register** and select **Pagelet Producer**.
5. Enter the connection details for Pagelet Producer (Table 22–2).

Table 22–2 Pagelet Producer Connection Parameters

Field	Description
Producer Name	A unique name to identify this Pagelet Producer instance within WebCenter Portal.

Server URL	The URL to the Pagelet Producer. The URL must include a fully-qualified domain name. Use the following syntax:
	`<protocol>://<host_name>:<port_number>/pagelets/` where host and port correspond to the `WC_Portlet` managed server where the Pagelet Producer is configured.
	For example:
	`http://myhost.com:7778/pagelets/`
	If pagelets contain secure data, the registered URL must use the `HTTPS` protocol. For example:
	`https://myhost.com:7779/pagelets/`
	The context root can be changed from /pagelets/ if necessary; for details, see Section 22.2.5, "Redeploying Pagelet Producer to a Different Context."
	Note: In WebCenter Portal, if the Pagelet Producer URL is protected by OAM, the URL to the pagelet catalog must be excluded (mapped directly without access control), or the catalog will appear to be empty when using REST. The pagelet catalog URL is `http://<host_name>:<port_number>/pagelets/api/v2/ensemble/pagelets`

22.2.5 Redeploying Pagelet Producer to a Different Context

In some cases, the default web context defined for the Pagelet Producer may need to be changed. This section describes how to redeploy the Pagelet Producer to a different context.

The first step is to target the Pagelet Producer data source to the Administration Server and locate the Pagelet Producer EAR file. In Oracle WebLogic Server:

1. In the Oracle WebLogic Server Console, go to **Services** > **Data Source**.
2. Click the **mds-PageletProducerDS** data source.
3. Go to the **Targets** tab.
4. Check the box next to **AdminServer** and click **Save**.
5. Navigate to Deployments/pagelet-producer.
6. If Fusion Middleware Control is running on the same host as Pagelet Producer, record the path to the EAR file. If Fusion Middleware Control is on a different host than Pagelet Producer, copy the EAR file from the Pagelet Producer host machine to the browser host machine.

Next, use Fusion Middleware Control to redefine the context:

1. Navigate to (Application) Deployments/pagelet-producer.
2. From the Application Deployment Menu, select **Application Deployment** > **Undeploy** and follow any prompts that appear. Click **Undeploy**.
3. From the Weblogic Domain menu, select **Application Deployment** > **Deploy**.

4. Set the Archive location to the Pagelet Producer EAR file (located and/or copied in the first set of steps above).
   - If Fusion Middleware Control is running on the same host as the Pagelet Producer, select the second option and browse to the EAR file location.
   - If Fusion Middleware Control is on a different host than Pagelet Producer, select the first option and click **Choose File** to select the EAR file from the location it was copied to on the browser host machine.
5. Select **WC_Portlet**.
6. Change the **Context Root** of the Web Modules as follows, where "new_context" is the Web context that should be used (to redeploy to root, omit "new_context"):
   - ensemblestatic.war: new_context/ensemblestatic
   - pageletadmin.war: new_context/admin
   - opensocial.war: new_context/os
   - loginserver.war: new_context/loginserver
   - ensembleproxy.war: new_context/

   Note: OpenSocial pagelets will not function properly if the Pagelet Producer is deployed to root context.
7. Click **Deploy**.

If your implementation uses OpenSocial, update the context setting in the Pagelet Producer Console. For details, see "Creating Pagelets with Pagelet Producer" section in the *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

22.3 Registering WSRP and Oracle JPDK Portlet Producers in the Pagelet Producer

The Pagelet Producer can expose WSRP and Oracle JPDK portlets as pagelets for use in Portal Framework applications, WebCenter Portal, and third-party portals.

You can use Fusion Middleware Control, WLST or the Pagelet Producer Console to register a WSRP or Oracle JPDK endpoint as a portlet producer. After registration, a new Pagelet Producer resource is created and automatically populated with pagelets to represent the portlets associated with the WSRP endpoint.

For detailed instructions, see Chapter 21, "Managing Portlet Producers." To access portlet producer settings from the Pagelet Producer Console, select **Producers** from the menu in the Navigator toolbar, then click **Register**.

22.3.1 Using WSRP and Oracle JPDK Portlets

Auto-generated WSRP and Oracle JPDK resources and pagelets cannot be modified. To make changes and create a permanent reference to the producer, the auto-generated resource must first be copied. Select the resource on the Shared Assets page and select **Copy** from the Action menu. The copied version of the resource can be edited, and various elements such as injectors can be added to customize pagelet functionality. Any replicated resources will be included in metadata exports.

You can also define a portlet-based pagelet from scratch by creating a new resource based on an existing portlet producer and then creating individual pagelets. For details, see the "Creating Pagelets with Pagelet Producer" section in the *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

22.4 Configuring the Trust Service Identity Asserter

This section describes how to configure the trust service identity asserter and includes the following subsections:

- Section 22.4.1, "About the Trust Service Identity Asserter"
- Section 22.4.2, "Preparing for Configuring the Trust Service Identity Asserter"
- Section 22.4.3, "Executing Trust Service Identity Asserter Configuration"

22.4.1 About the Trust Service Identity Asserter

The WebCenter Portal communicates with a Pagelet Producer using a server to server REST call. In order to pass the identity of the administrative user to the Pagelet Producer a WLS "Trust Service Identity Asserter" must be set up on the Pagelet Producer (server) and OPSS keystore service credentials must be set up on both the Pagelet Producer (server) and WebCenter Portal (client). For more information, see "Integrating Application Security with OPSS" in *Oracle Fusion Middleware Application Security Guide*.

22.4.2 Preparing for Configuring the Trust Service Identity Asserter

The WebCenter Portal installation (same installer is used for both the WebCenter Portal and the Pagelet Producer) will place the following two files in the *WCP_HOME*`/webcenter/scripts` directory (for example, `/home/user/Oracle/Middleware/Oracle_WC1/webcenter/scripts`):

- `configureTrustServiceIdentityAsserter.py`
- `configureTrustServiceIdentityAsserter.properties`

The WLST script `configureTrustServiceIdentityAsserter.py` uses the values set in the `configureTrustServiceIdentityAsserter.properties` file to configure trust identity on both the client (WebCenter Portal) and server (Pagelet Producer).

Properties to Fill Out

The following properties must be filled out before executing `configureTrustServiceIdentityAsserter.py`:

Table 22–3 Properties Used by configureTrustServiceIdentityAsserter.py

Property	Description	Example Value
`admin.user`	WLS administrative user	`weblogic`
`admin.password`	WLS administrative user password	`welcome1`
`admin.url`	WLS administrative server host url	`t3://localhost:7001`
`trust.alias`	Keystore alias name that will contain private key pair used for signing token used in REST calls. Use alphanumeric characters.	`wckey`

Table 22–3 (Cont.) Properties Used by configureTrustServiceIdentityAsserter.py

Property	Description	Example Value
`trust.issuer`	This is the value placed inside the token that indicates who the issuer of the token is	`mycompany`
`keystore.exported.cert`	This is a file path where the public key for the key pair in trust.alias is exported to and exported from. For information about creating this certificate, see the "SAML Security Between a WebCenter Portal: Framework Application Consumer and a WebLogic Portal Producer" section in *Oracle Fusion Middleware Federated Portals Guide for Oracle WebLogic Portal*.	`/home/user/Oracle/Middleware/user_projects/domains/my_domain/config/fmwconfig/wckey.cer`

In addition to the above properties there are several optional properties defined in `configureTrustServiceIdentityAsserter.properties`. If these properties are not defined in the file the values listed under 'Default Value' column below will be used:

Table 22–4 Properties Used by configureTrustServiceIdentityAsserter.py

Original Property	Description	Default Value
`keystore.distinguished.name`	DN used in keystore key pair generation	`CN=<property value of trust.issuer>,O=Oracle,C=US`
`trust.identity.asserter.name`	Name to give the WLS Trust Service Identity Asserter	`TrustServiceIA`

For more details, open the `configureTrustServiceIdentityAsserter.properties` file. A full description of each property and the overall trust identity assertion configuration process is provided in inline comments.

22.4.3 Executing Trust Service Identity Asserter Configuration

WebCenter Portal and Pagelet Producer on same WLS Domain

In most deployment scenarios, the Pagelet Producer and WebCenter Portal run on separate WebLogic managed servers on the same WebLogic domain. In this scenario, the OPSS keystore configuration runs once and handles both the client (WebCenter Portal) and server (Pagelet Producer) set up as shown in the following examples:

```
cd WCP_ORACLE_HOME/webcenter/scripts
WCP_ORACLE_HOME/common/bin/wlst.sh ./configureTrustServiceIdentityAsserter.py
./configureTrustServiceIdentityAsserter.properties
```

Note that for Windows environments the `.sh` is not needed.

22.5 Managing Import, Export, Backup and Recovery of Pagelet Producer Components

Pagelet Producer stores data related to its configuration and content in the Oracle metadata store (MDS) to facilitate disaster recovery and the full production lifecycle from development through staging and production. This section describes the import, export and backup capabilities available.

- Section 22.5.1, "Exporting and Importing Pagelet Producer Resources"
- Section 22.5.2, "Exporting and Importing Pagelet Producer Metadata Using WLST"
- Section 22.5.3, "Backing Up and Restoring the Pagelet Producer"

For detailed information about MDS, see the chapter "Managing the Oracle Metadata Repository" in the *Oracle Fusion Middleware Administrator's Guide*.

22.5.1 Exporting and Importing Pagelet Producer Resources

Pagelet Producer assets can be exported and imported using the Pagelet Producer Console. Note that you cannot export or import pagelets directly from the Shared Assets page in WebCenter Portal. To export or import Pagelet Producer shared assets you must use the Pagelet Producer Console as described in this section, or use WLST as described in Section 22.5.2, "Exporting and Importing Pagelet Producer Metadata Using WLST."

Note: For WAS environments only, you must map the `orcladmin` user to all of the following roles: `Anonymous`, `EnsembleAdmin`, and `Admin`. If the `orcladmin` user is not mapped to one of these roles the user interface may not render correctly.

To import or export Pagelet Producer assets using the Pagelet Producer Console:

1. Open the Pagelet Producer Console.

   You can do this by either:

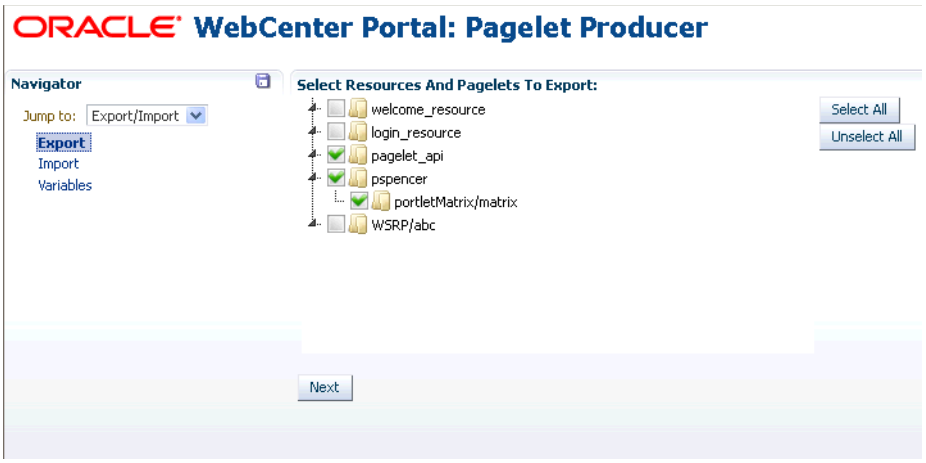
   - From WebCenter Portal, navigating to **Administration > Shared Assets > Pagelets** and then clicking **Create**. This opens the Pagelet Producer Console. When you're ready to return to WebCenter Portal click **Cancel**.
   - Navigating to the following URL:

     `http://<host_name>:<port_number>/pagelets/admin`.

2. From the **Jump to:** dropdown list, select **Export/Import**.
3. Click either **Export**, **Import**, or **Variables** to select the activity to be performed:
   - Use the **Export** pane to choose from a list of assets and export them to a new MDS package.
   - Use the **Import** pane to browse to an existing MDS package and import it into the Pagelet Producer.
   - Use the **Variables** pane to define variables for root URLs to protect internal URLs and simplify import.
4. To export resources, click **Export**.

   The Export pane displays (see Figure 22–1)

Figure 22–1 Pagelet Producer Console - Export Pane



a. Check the items to include in the export.

b. Click **Next**.

The Host URL displays (Figure 22–2):

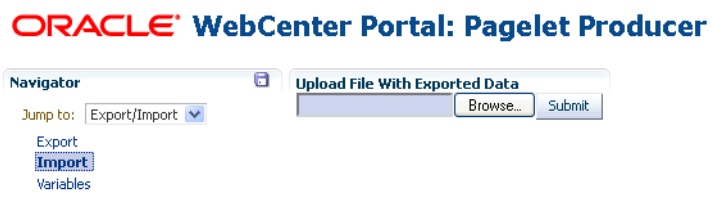
Figure 22–2 Host URL

c. Enter the URL for the **Host** (click the **Variable** field to use a variable if you've defined one) and then click **Export**.

5. To import resources, click **Import**.

The Import pane displays (see Figure 22–1).

Figure 22–3 Pagelet Producer Console - Import Options



a. Click **Browse** to select the file to import.

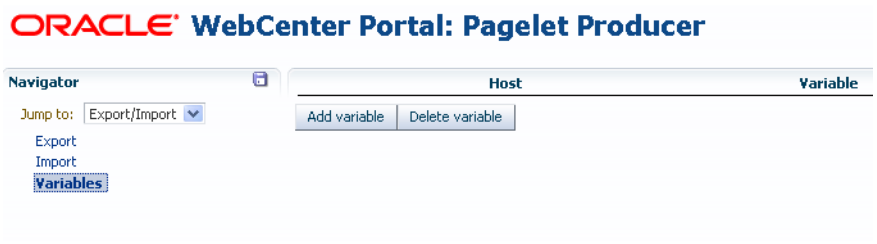
b. Click **Submit** to start the import.

c. If prompted, select either **Skip** or **Overwrite** if there is an existing resource on the target side of the import.

6. To define a variable, click **Variables**.

The Variables pane displays (Figure 22–4).

Figure 22–4 Pagelet Producer Console - Variables Pane



a. Click **Add Variable.**

b. Enter the host name in the **Host** field.

c. Enter the variable name with which to associate the host URL in the **Variable** field.

d. To continue adding variables, click **Add Variable**.

   Once added, you can use the variables as part of the host URL in the Export pane.

22.5.2 Exporting and Importing Pagelet Producer Metadata Using WLST

The metadata created by the Pagelet Producer is stored in MDS and can be accessed using WLST. For detailed information on running WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Only global migration using WLST is currently supported; all data in the source environment is included in the exported MDS package, and all data in the target environment is overwritten when the package is imported.

> **Note:** If you are migrating your WebCenter Portal implementation from staging to production, exporting and importing Pagelet Producer data is handled by the migration tool. However, if changes were made to Pagelet Producer objects in the staging environment, these changes must be migrated independently using the WLST commands described in this section. If the Pagelet Producer does not function after migration, check the Server URL defined for the Pagelet Producer in your WebCenter Portal application. For information on setting this URL, see Section 22.2, "Registering Pagelet Producer." For details on WebCenter Portal migration, see Chapter 39, "Understanding WebCenter Portal Life Cycle."

22.5.2.1 Exporting Pagelet Producer Metadata Using WLST

To export base documents for the Pagelet Producer, including any resources, pagelets and custom configuration settings, use the WLST command `exportMetadata`.

For example:

```
exportMetadata(application='pagelet-producer', server='WC_Portlet_Staging',
toLocation='c:\work\myexport', docs='/**')
```

Where:

- `application`: Name of the Pagelet Producer application for which the metadata is to be exported (for example, `pagelet-producer`).

- `server`: Server on which the Pagelet Producer is deployed (for example, `WC_PortletStaging`).
- `toLocation`: Target directory to which documents selected from the source partition are to be exported. The `toLocation` parameter can be used as a temporary file system for migrating metadata from one server to another.
- `docs`: List of comma-separated fully qualified document name(s) and/or document name patterns (* and ** patterns).

For detailed syntax and examples, see "exportMetadata" in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

22.5.2.2 Importing Pagelet Producer Metadata Using WLST

To import Pagelet Producer metadata and customizations, use the WLST command `importMetadata`.

For example:

```
importMetadata(application='pagelet-producer', server='WC_Portlet_Production',
fromLocation='c:\work\myexport', docs='/**')
```

Where:

- `application`: Name of the Pagelet Producer application for which the metadata is be imported (for example, `pagelet-producer`).
- `server`: Name of the target server on which the Pagelet Producer is deployed (for example, `WC_Portlet_Production`).
- `fromLocation`: Source directory from which documents are imported. The `fromLocation` parameter can be any temporary file system location for migrating metadata from one server to another.
- `docs`: List of comma separated fully qualified document name(s) and/or document name patterns (* and ** patterns).

For detailed syntax and examples, see "importMetadata" in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

> **Note:** Any environment-specific URLs used in object configuration must be updated manually after import.

22.5.3 Backing Up and Restoring the Pagelet Producer

Backup and recovery operations for the Pagelet Producer are part of standard MDS backup and restoration and can be managed through database export and import utilities, and various other tools. For detailed information, see "Part VII" in *Oracle Fusion Middleware Administrator's Guide*.

By default, the MDS configuration for Pagelet Producer is as follows (from `adf-config.xml`):

```
<metadata-store name="PageletProducerMetadataRepos"
class-name="oracle.mds.persistence.stores.db.DBMetadataStore">
  <property name="partition-name" value="pageletproducer"/>  <property
name="jndi-datasource" value="jdbc/mds/PageletProducerDS"/>
  <property name="repository-name" value="mds-PageletProducerDS"/>
</metadata-store>
```

23

Managing External Applications

An external application is any application that implements its own authentication process. Specifically, it is an application that does not take part the single sign-on process for your portal application (that is, WebCenter Portal or your own Portal Framework application).

System administrators can use Fusion Middleware Control or the WLST command-line tool to register and manage external applications for WebCenter Portal and Portal Framework application deployments.

Application administrators can also register and manage external applications at runtime through out-of-the-box administration pages or using external application task flows.

All external application changes that you make for WebCenter Portal or Portal Framework applications post deployment, are stored in the MDS repository as customizations.

> **Note:** External application configuration is dynamic. Configuration changes are immediately reflected in WebCenter Portal and Portal Framework applications; it is not necessary to restart the application or the managed server.

This chapter includes the following sections:

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console and the `Administrator` role in the deployed application:

- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.
- **Portal Framework application**: `Administrator` role granted through the Administration Console.

For more information about roles and permissions, see Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

23.1 About External Applications

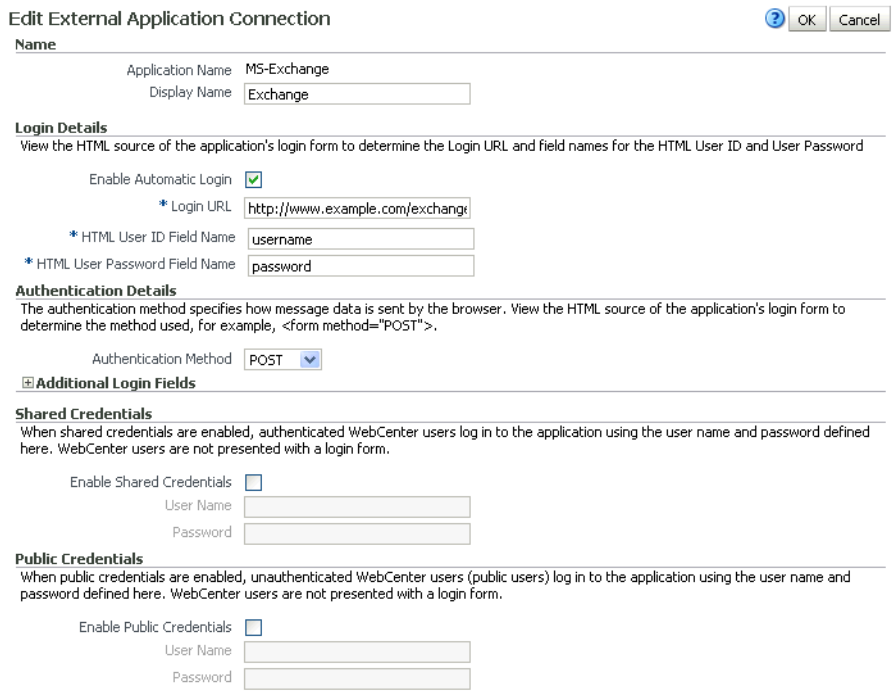
If WebCenter Portal your own Portal Framework application interacts with an application that handles its own authentication, you can associate that application with an external application definition to allow for credential provisioning. In doing so, you use an external application definition to provide a means of accessing content from these independently authenticated applications.

To replicate a single sign-on experience from the end user's perspective, the external application service captures the user name and password, and any other credentials for the external application, and supplies it to the WebCenter Portal tool or application requiring the credentials. The WebCenter Portal tool or other application then uses this information to log in on behalf of the end user. This username and password combination is securely stored in a credential store configured for the WebLogic domain where the application is deployed.

Note: When logging in to an external application, if you clear the **Remember My Login Information** check box, then the credentials provisioned for that user session are lost in the event of a failover in a high availability (HA) environment. You are prompted to specify the credentials again if you try to access the external application content in the same user session.

The external applications that are to be used by a Portal Framework application can be specified before deployment through a wizard in Oracle JDeveloper, or after deployment through Fusion Middleware Control Console (Figure 23–1) or using WLST commands. Post-deployment, external applications specified at design time in JDeveloper display automatically. However, after deployment you must reprovision design-time shared and public credentials using Fusion Middleware Control or WLST commands. For information, see Chapter 31, "Configuring the Identity Store," and Chapter 32, "Configuring the Policy and Credential Store."

Figure 23–1 Edit External Application



23.2 Registering External Applications

You can register external applications for WebCenter Portal and Portal Framework applications through Fusion Middleware Control or using WLST commands.

Before registering an external application, access the application's login page and examine the HTML source for the application's login form. All the registration details you require are located in the `<form tag>`.

For example, the underlying code for the *Yahoo! Mail* login form looks something like this:

```
<form method=post action="https://login.yahoo.com/config/login?"
autocomplete="off" name="login_form">
...
<td><input name="login" size="17"</td>
...
<td><input name="passwd" size="17"</td>
...
```

In this example, to provide WebCenter Portal users with a direct link to the *Yahoo! Mail* application, the following sample registration information is required:

Registration Information	Sample Value	HTML Source
Login URL	`https://login.yahoo.com/config/login?`	`action`
User Name / User ID Field	`login`	`name="login"`
Password Field Name:	`passwd`	`name="passwd"`
Authentication Method	`post`	`method`

> **Note:** External application configuration is dynamic. New external applications and updates to existing applications are immediately available; there is no need to restart WebCenter Portal or your Portal Framework application.

For information about tools that use external applications, see the "Secured Service Connections" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

This section includes the steps for:

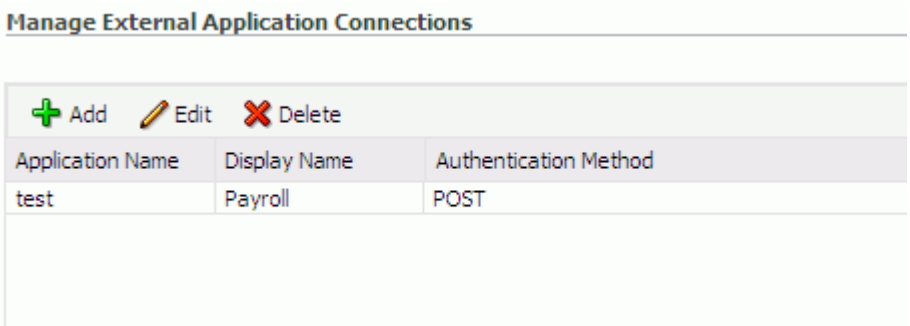
- Section 23.2.1, "Registering External Applications Using Fusion Middleware Control"
- Section 23.2.2, "Registering External Applications Using WLST"
- Section 23.2.3, "Registering External Applications in WebCenter Portal"
- Section 23.2.4, "Registering External Applications in Portal Framework Applications"

23.2.1 Registering External Applications Using Fusion Middleware Control

To register an external application:

1. Login to Fusion Middleware Control and navigate to the home page for your WebCenter Portal or Portal Framework application:
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.
3. From the list of services on the **WebCenter Portal Service Configuration** page, select **External Applications**.
4. To register a new external application, click **Add** (Figure 23–2).

Figure 23–2 Configuring External Application Connections



5. Enter a unique name for the external application and a display name that application users working with this external application will see.

   See also Table 23–1.

Table 23–1 External Application Connection - Name

Field	Description
Application Name	Enter a name for the application. The name must be unique (across all connection types) within the application. For example: `yahoo` **Note:** Once registered, you cannot edit the Application Name.
Display Name	Enter a user friendly name for the application that WebCenter Portal users will recognize. Application end-users working with this external application will see the display name you specify here. For example: `My Yahoo` If you leave this field blank, the Application Name is used.

6. Enter login details for the external application.

   For details, see Table 23–2.

Table 23–2 External Application Connection - Login Details

Field	Description
Enable Automatic Login	Select to allow automatically log users in to this application. Choosing this option requires you to complete the `Login URL`, `HTML User ID Field Name`, and `HTML User Password Field Name` fields With automated single sign-on, the user directly links to the application and is authenticated automatically, as their credentials are retrieved from the credential store. Selecting this option provides the end user with a seamless single sign-on experience. **Note:** Automated login is not supported for: ▪ External applications using BASIC authentication. ▪ External applications configured for SSO. ▪ External applications with a customized login form (built using ADF Faces) that does not implement the J2EE security container login method `j_security_check` for authentication. ▪ External sites that do not support UTF8 encoding. ▪ External applications that accept randomly generated hidden field values or cookies for successful login.
Login URL	Enter the login URL for the external application. To determine the URL, navigate to the application's login page and record the URL. For example: `http://login.yahoo.com/config/login` **Note:** A login URL is not required if the sole purpose of this external application is to store and supply user credentials on behalf of another service.

Table 23–2 (Cont.) External Application Connection - Login Details

Field	Description
HTML User ID Field Name	Enter the name that identifies the "user name" or "user ID" field on the login form. **Tip:** To find this name, look at the HTML source for the login page. This property does not specify user credentials. Mandatory if the **Authentication Method** is GET or POST. Leave this field blank if the application uses BASIC authentication (see **Authentication Method**).
HTML User Password Field Name	Enter the name that identifies the "password" field on the login form. **Tip:** To find this name, look at the HTML source for the login page. Mandatory if the **Authentication Method** is GET or POST. Leave this field blank if the application uses BASIC authentication (see **Authentication Method**).

7. Select the authentication method used by the external application.

   For details, see Table 23–3.

Table 23–3 External Application Connection - Authentication Details

Field	Description
Authentication Method	Select the form submission method used by the external application. Choose from one of the following: ▪ **GET**: Presents a page request to a server, submitting the login credentials as part of the login URL. This authentication method may pose a security risk because the user name and password are exposed in the URL. ▪ **POST**: Submits login credentials within the body of the form. This is the default. ▪ **BASIC**: Submits login credentials to the server as an authentication header in the request. This authentication method may pose a security risk because the credentials can be intercepted easily and this scheme also provides no protection for the information passed back from the server. The assumption is that the connection between the client and server computers is secure and can be trusted. The **Authentication Method** specifies how message data is sent by the browser. You can find this value by viewing the HTML source for the external application's login form, for example, `<form method="POST" action="https://login.yahoo.com/config/login?" AutoComplete="off">`

8. Specify additional login fields and details, if required.

   For details, see Table 23–4, " External Application Connection - Additional Login Fields".

Table 23–4 External Application Connection - Additional Login Fields

Field	Description
Additional Login Fields	If your application requires additional login criteria, expand **Additional Login Fields**. For example, in addition to *user name* and *password*, the Lotus Notes application requires two additional fields - *Host* and *MailFilename*. Click **Add** to specify an additional field for the login form. For each new field, do the following: ■ **Name** - Enter the name that identifies the field on the HTML login form that may require user input to log in. This field is not applicable if the application uses basic authentication. ■ **Value** - Enter a default value for the field or leave blank for a user to specify. This field is not applicable if the application uses basic authentication. ■ **Display to User** - Select to display the field on the external application login screen. If the field is not displayed (unchecked), then a default **Value** must be specified. Click **Delete** to remove a login field.

9. Specify shared and public user credentials, if required.

For details, see Table 23–5.

Table 23–5 External Application Connection - Shared User and Public User Credentials

Field	Description
Enable Shared Credentials	Indicate whether this external application enables shared user credentials, and specify the credentials. Select **Enable Shared Credentials**, and then enter **User Name** and **Password** credentials for the shared user. When shared credentials are specified, every user accessing this external application, through either WebCenter Portal or your Portal Framework application, is authenticated using the user name and password defined here. WebCenter Portal users are not presented with a login form. Because WebCenter Portal users do not need to define personal credentials of their own, external applications with shared credentials are not listed in the external application's change password task flows such as *My Accounts*. See also the "Providing Login Information for External Applications" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.
Enable Public Credentials	Indicate whether unauthenticated users (public users) may access this external application. Select **Enable Public Credentials**, and then enter **User Name** and **Password** credentials for the public user. When public credentials are specified, public users accessing this external application through either WebCenter Portal or your Portal Framework application's public pages, are logged in using the username and password defined here. If public credentials are not specified, public users will see an authorization error indicating this external application is not accessible to public users.

10. Click **OK** to register the application.

23.2.2 Registering External Applications Using WLST

Use the WLST command `createExtAppConnection` to create an external application connection. For command syntax and examples, see `createExtAppConnection` in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

Use the WLST command `addExtAppCredential` to add shared or public credentials for an existing external application connection. For details, see `addExtAppCredential` in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

Use the WLST command `addExtAppField` to define additional login criteria for an existing external application connection. For details, see `addExtAppField` in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

23.2.3 Registering External Applications in WebCenter Portal

For information about registering external applications through Portal Builder Administration, see Section 8.3.2, "Configuring Tools and Services for WebCenter Portal".

23.2.4 Registering External Applications in Portal Framework Applications

For information about registering external applications in Portal Framework applications, see Section 23.4, "Managing External Applications with the WebCenter Portal Administration Console".

23.3 Modifying External Application Connection Details

This section shows you how to modify the external application connection details by:

- Section 23.3.1, "Modifying External Application Connection Using Fusion Middleware Control"
- Section 23.3.2, "Modifying External Application Connection Using WLST"

23.3.1 Modifying External Application Connection Using Fusion Middleware Control

To update external application connection details:

1. Log in to Fusion Middleware Control and navigate to the home page for your WebCenter Portal or Portal Framework application:
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications".
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.

3. From the list of services on the **WebCenter Portal Service Configuration** page, select **External Applications**.
4. Select the name of the external application you want to modify, and click **Edit**.
5. Edit connection details, as required. For detailed parameter information, see Table 23–2.

   Note that you cannot edit the name of the external application.
6. Click **OK** to save your changes.

23.3.2 Modifying External Application Connection Using WLST

Use the WLST command `setExtAppConnection` to edit existing external application connection details. For command syntax and examples, see `setExtAppConnection` in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

Note: To edit details relating to an additional login field, use `setExtAppField`. To edit existing shared or public credentials, use `setExtAppCredential`.

To delete an additional login field, use `removeExtAppField`. To delete shared or public credentials, use `removeExtAppCredential`.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

For information about modifying external applications in WebCenter Portal, see the "Editing External Application Connection Details" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

23.4 Managing External Applications with the WebCenter Portal Administration Console

To be able to register and manage external applications in a Portal Framework application, a user must be assigned the `AppConnectionManager` role. By default, users with the `Administrator` role have the `AppConnectionManager` role; and therefore, application administrators can configure external applications through the WebCenter Portal Administration Console.

An external application is any application that implements its own authentication process. Specifically, it is an application that does not take part in the Portal Framework application's single sign-on process. If your Portal Framework application interacts with an application that handles its own authentication, you can register an external application to allow for credential provisioning.

Application administrators can register, edit, and delete external applications for a Portal Framework application at runtime, through the WebCenter Portal Administration Console.

This section includes the following subsections:

- Section 23.4.1, "Registering External Applications"
- Section 23.4.2, "Editing and Deleting External Applications"

Note: System administrators can also register external applications for Portal Framework applications, using Fusion Middleware Control and WLST commands. For details, see Chapter 21, "Managing Portlet Producers."

23.4.1 Registering External Applications

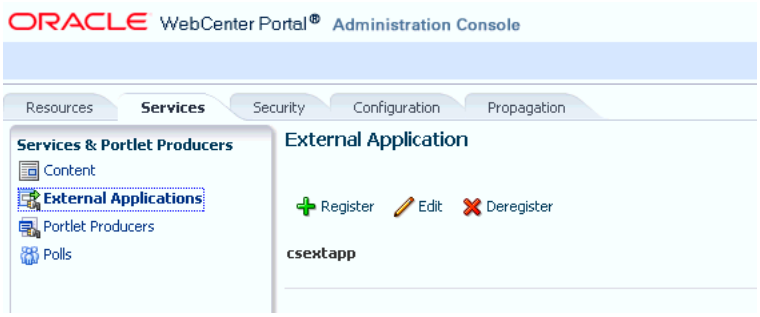
To register an external application at runtime for a Portal Framework application:

1. Navigate to the **Services** administration tab.

   See also, Section 43.2, "Accessing the Administration Console for Portal Framework Applications".

2. Select **External Application** (Figure 23–3).

Figure 23–3 WebCenter Portal Administration Console - External Applications



3. Click **Register**.
4. Enter connection details for the external application.

   If you need help with one or more fields, refer to the following tables:

   - Table 23–1, " External Application Connection - Name"
   - Table 23–2, " External Application Connection - Login Details"
   - Table 23–3, " External Application Connection - Authentication Details"
   - Table 23–4, " External Application Connection - Additional Login Fields"
   - Table 23–5, " External Application Connection - Shared User and Public User Credentials"

5. Click **Test** to verify your connection details.
6. Click **OK** to register the application.

23.4.2 Editing and Deleting External Applications

To modify or delete external applications at runtime for a Portal Framework application:

1. Navigate to the **Services** administration tab.

   See also, Section 43.2, "Accessing the Administration Console for Portal Framework Applications".

2. Select **External Application** (Figure 23–3).

3. Select the external application required and then click one of the following:
   - Click **Edit** to update connection details for an external application.
   - Click **Deregister** to remove the external application.

     Take care when deleting an external application connection as Portal Framework application users will no longer have access to that application, and any services dependent on the external application may not function correctly.

23.5 Testing External Application Connections

For external applications that are created using login URLs, ensure that their login URLs are accessible. For information about direct URLs, see the "Automated Single Sign-On" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

23.6 Deleting External Application Connections

Take care when deleting an external application connection as users in WebCenter Portal or your Portal Framework application will no longer have access to that external application, and any tools or services dependent on the external application may not function correctly.

This section includes the following subsections:

- Section 23.6.1, "Deleting External Application Connections Using Fusion Middleware Control"
- Section 23.6.2, "Deleting External Application Connections Using WLST"

23.6.1 Deleting External Application Connections Using Fusion Middleware Control

To delete an external application connection:

1. Log into Fusion Middleware Control and navigate to the home page for your WebCenter Portal or Portal Framework application:
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Service Configuration**.
3. From the list of services on the **WebCenter Portal Service Configuration** page, select **External Applications**.
4. Select the name of the external application you want to remove, and click **Delete**.

23.6.2 Deleting External Application Connections Using WLST

Use the WLST command `deleteConnection` to remove an external application connection. For command syntax and examples, see `deleteConnection` in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

Note: To delete an additional login field, use `removeExtAppField`. To delete shared or public credentials, use `removeExtAppCredential`.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

23.7 Troubleshooting External Application Issues

This section contains common issues and workarounds related to external applications.

This section contains the following subsections:

- Section 23.7.1, "Users Experience Password Lockout"

23.7.1 Users Experience Password Lockout

Problem

Using an external application to store or retrieve credentials for collaboration connections when your identity store uses a password change policy that causes the password to be changed in the identity store directly, may lead users to experience a password lockout.

Solution

The external applications cannot know that a password has been changed directly in the identity store and consequently cannot react to it. A partial solution is to define one external application for all your collaboration connections. For releases prior to PS6, contact support to apply patches for bugs 9327220, 12965480, 14174484, 14309006 in your environment.

24

Managing REST Services

This chapter provides an overview of managing Oracle WebCenter Portal's REST services in WebCenter Portal or Portal Framework applications.

This chapter includes the following sections:

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console and the `Administrator` role in the deployed application:

- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.
- **Portal Framework application**: `Administrator` role granted through the Administration Console.

For more information about roles and permissions, see Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

24.1 About REST Services

REST (REpresentational State Transfer) is an architectural style for making distributed resources available through a uniform interface that includes uniform resource identifiers (URIs), well-defined operations, hypermedia links, and a constrained set of media types. Typically, these operations include reading, writing, editing, and removing. Media types include JSON and XML/ATOM.

REST APIs are commonly used in client-side scripted, Rich Internet Applications. For example, a browser-based application written in Javascript can use Ajax techniques with REST APIs to send and receive application data from the server and update the client view.

Oracle WebCenter Portal provides a RESTful interface to many of its tools and services, like discussions, lists, people connections, and search. For a complete list of the services that support REST, see "Using Oracle WebCenter Portal REST APIs" in the

Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper.

For a more complete introduction to REST and Oracle WebCenter Portal REST APIs, see "Using Oracle WebCenter REST APIs" in the *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

24.2 Performing Required Manual Configurations to Enable REST

Oracle WebCenter Portal REST APIs are not enabled by default. To enable the REST APIs to work, you must perform the two separate server-side configurations: you must configure an identity asserter and you must seed required entries in the credential store to enable the REST security tokens to function properly. For more information on security tokens, see "Security Considerations for WebCenter Portal REST APIs" in the *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

Perform the following configuration tasks after Oracle WebCenter Portal is installed for the first time or if you know the configuration tasks have not been previously performed.

- Section 24.2.1, "Configuring an Identity Asserter"
- Section 24.2.2, "Configuring the WebLogic Server Credential Store"

24.2.1 Configuring an Identity Asserter

You must configure an identity asserter before using the REST APIs. For detailed instructions, see Section 31.9, "Configuring the REST Service Identity Asserter."

24.2.2 Configuring the WebLogic Server Credential Store

After configuring an identity asserter, the next step is to configure the WLS credential store. To configure the credential store, execute the following WLST commands while the server is running. No restart is required.

```
createCred(map="o.webcenter.jf.csf.map", key="keygen.algorithm",
    user="keygen.algorithm", password="AES")
createCred(map="o.webcenter.jf.csf.map", key="cipher.transformation",
    user="cipher.transformation", password="AES/CBC/PKCS5Padding")
```

24.3 Understanding Security Tokens

A user-scoped security token is embedded in the `href` and `template` attributes of every REST service URI. The token is both generated and validated by the server, and is enabled by the `keygen.algorithm` and `cipher.transformation` configuration steps described in Section 24.2.2, "Configuring the WebLogic Server Credential Store." The purpose of the security token is to prevent Cross-Site Request Forgery (CSRF) attacks.

For example:

```
<link
  template="opaque-template-uri/@me?startIndex={startIndex}
    &itemsPerPage={itemsPerPage}&utoken=generated-token"
  resourceType="urn:oracle:webcenter:messageBoard"
  href="opaque-uri/@me?token=generated-token"
  capabilities="urn:oracle:webcenter:read"/>
```

Note: The security token is not used for authentication or identity propagation.

Security tokens are based on the authenticated user's name. They do not expire, making it possible to both cache and bookmark the URIs.

Security tokens are also "salted," a cryptographic technique of adding extra characters to a string before encrypting it. Because of salting, if a security token is compromised, you will not have to change the user's user name across the entire system to address the problem.

This technique prevents cases where a user name is compromised and you don't want to have to change the user name system wide to fix the problem. If you need to regenerate the salt, you can do so by simply deleting it with the following WLST command:

```
deleteCred(map="o.webcenter.jf.csf.map", key="user.token.salt", user="
user.token.salt", password="AES")
```

For more information on security tokens, see "Security Considerations for WebCenter Portal REST APIs" in the *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

24.4 Changing the REST Root Name

Although not required, in some cases you might want to change the root name for the REST APIs. The recommended technique for changing the REST root name is to do so by configuring a proxy server as described in Section 8.2.4, "Setting Up a Proxy Server." For example, through the proxy server configuration, the following two URIs refer to the same server:

`http://myhost:8888/rest/api/resourceIndex`

`http://myhost:8888/pathname/rest/api/resourceIndex`

24.5 Using Compression

This section explains techniques for enabling compression on the XML or JSON responses that are returned to the client by the Oracle WebCenter Portal REST APIs.

If you are running Apache, you can add the `mod_deflate` or `mod_gzip` server modules to the server configuration. Refer to the Apache documentation for more information.

If you are using Oracle HTTP Server (OHS), Oracle recommends using Oracle Web Cache for this purpose. For detailed information, see *Oracle Fusion Middleware Administrator's Guide for Oracle Web Cache*.

If you are using OHS, you can also add the `mod_deflate` or `mod_gzip` server module to enable compression. For detailed information on this technique, see "Understanding Oracle HTTP Server Modules" in the *Oracle Fusion Middleware Administrators Guide for Oracle HTTP Server*.

For more information on Oracle Web Cache, see the section "Compression" in the *Oracle Fusion Middleware Administrators Guide for Oracle HTTP Server*, and see the chapter "Caching and Compressing Content" in the same guide.

24.6 Handling Authentication

By default, REST services are configured to accept authentication from identity assertion providers. If no identity assertion providers are configured, basic authentication is used.

For information on configuring identity assertion providers, see Section 31.9, "Configuring the REST Service Identity Asserter."

For more information, see "Configuring Authentication Providers" in *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

Note: Users can access the CMIS root anonymously. For more information, see "Security Considerations for CMIS REST APIs" in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

25

Managing Personalization

This chapter describes how to configure and manage personalization in Oracle WebCenter Portal.

This chapter includes the following sections:

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console and the `Administrator` role in the deployed application:

- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.
- **Portal Framework application**: `Administrator` role granted through the Administration Console.

For more information about roles and permissions, see Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

25.1 About Personalization for Oracle WebCenter Portal

Personalization for Oracle WebCenter Portal provides a dynamically derived user experience for your portal application. Personalization evaluates defined sources of input data, generates a decision based on that evaluation, and applies this information to a declaratively defined personalization scenario. Personalization, for example, can return content or change application flow based on information about a user in a Human Resources database, targeting the application experience for that specific user.

Personalization is installed as an application in the `wc_domain` on the `WC_Utilities` server. Client applications access Personalization remotely over HTTP

using RESTful services. Design-time JDeveloper tools are used to create Property Service and Conductor artifacts to be executed remotely using REST calls.

Personalization is also available in the JDeveloper integrated domain for projects that include the Personalization technology libraries when you first create your application. For evaluation purposes and iterative development, this domain offers the quickest and easiest way to explore Personalization. For more information about the Personalization architecture and services, see the "Personalizing Oracle WebCenter Portal Applications" chapter in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

25.2 Personalization Prerequisites and Limitations

This section describes the system requirements and dependencies for Personalization in the following sections:

- Section 25.2.1, "Personalization Installation Requirements"
- Section 25.2.3, "Personalization Configuration Requirements"
- Section 25.2.4, "Personalization Security"
- Section 25.2.5, "Personalization Limitations"
- Section 25.2.6, "Personalization Configuration Options"

25.2.1 Personalization Installation Requirements

If you are using the CMIS (Content Management Interoperability Services) or Activity Graph data providers, or the People Connections locator within a Personalization Conductor scenario, then Oracle WebCenter Portal must be installed. For High Availability environments only, Coherence is also required.

25.2.2 Personalization REST API Configuration Requirements

Before you can use the Personalization REST APIs, you must perform two server-side configurations described in Section 24.2, "Performing Required Manual Configurations to Enable REST." For more information on security tokens, see also the "Security Considerations for WebCenter Portal REST APIs" section in the *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

25.2.3 Personalization Configuration Requirements

If you are using the CMIS provider, Activity Graph provider, PeopleConnections locator, or custom providers you must configure them as shown in Section 25.4, "Configuring Providers."

If you are using Content Presenter to present content in a portal or Portal Framework application, then you must also configure Content Presenter to display the results of your scenarios as described in Section 25.6, "Configuring Content Presenter."

Personalization relies on Trust Services to provide single sign-on (SSO) between different managed servers within the WebCenter Portal domain. Trust Services must be configured using the WLST scripts `configureTrustWCPS.py` and `configureConnectionsWCPS.py` provided in the `user_projects/applications/wc_domain` directory. For JDeveloper's integrated domain, only a single script (`configureWCPS.py`) located in the `DefaultDomain/scrpts-wcps` directory is used. For more information about configuring Trust Services and single sign-on using this script see Section 25.7,

"Configuring Single Sign-On."

25.2.4 Personalization Security

Personalization is compatible with whatever source of user authentication services are configured within the WLS security realm. That is, it can use the default identity store and policy and credential store for the domain.

If you are using the People Connections locator or Activity Graph data providers, users must also be configured as WebCenter Portal users.

Personalization REST services are accessed though a pre-configured Personalization Web application that requires authenticated access for all resources (all URIs), with the exception of the `resourceIndex`. You can modify these constraints to provide either less security to execute scenarios (where anonymous access is needed), or more security to prevent the ability to create new scenarios. For information about modifying the default security settings, see Section 25.8, "Overriding the Default Security Settings."

In WebCenter Portal, Trust Services provides single sign-on for Personalization REST calls. This requires that the WLS TrustServicesIdentityAsserter is configured (it is not pre-configured). You can do this manually using the WLS Console or with the provided WLST scripts `configureTrustWCPS.py` and `configureConnectionsWCPS.py` located in the `user_projects/applications/wc_domain` directory. For JDeveloper's integrated domain, a single script (`configureWCPS.py`) located in the `DefaultDomain/scrpts-wcps` directory is used.

You can also optionally secure your WebCenter Portal application's connection to the Personalization server and Personalization providers with single sign-on. For more information about configuring single sign-on, see section Section 25.7, "Configuring Single Sign-On." Access to Property Service data can also be limited by an application using a filter (IPropertyPermission) to pre-authorize access to property data.

Scenarios can use an out-of-the-box function library supporting basic Role evaluation and testing to authorize access to scenarios.

25.2.5 Personalization Limitations

By default, Personalization uses a managed server-scoped cache, meaning any changes made to cached data outside the managed server will not be seen by additional installations of Personalization.

For clustered (multiple) deployments of Personalization, Coherence may be configured for a cluster-aware cache.

25.2.6 Personalization Configuration Options

This section describes the out-of-the-box providers and other optional extensions to Personalization for WebCenter Portal, and the configuration required to integrate them into your Personalization project.

The out-of-the-box Personalization data providers allow you to write scenarios and access profile data based on existing WebCenter Portal tools. These WebCenter Portal tools expose their data via RESTful Web services. The Personalization data providers act as REST clients of these Web services and make it easy to author scenarios within JDeveloper based on these external data sources. You can also provide your own data provider and property locator implementations to integrate your own sources of external data.

CMIS Provider

The CMIS provider is an out-of-the-box provider that you can optionally use as a data source in your Personalization project. WebCenter Portal content services are exposed using the CMIS (Content Management Interoperability Services) standard. The CMIS REST service runs on the `WC_Spaces` server and provides access (based on separate configuration choices) to Oracle WebCenter Content Server.

If a Personalization user is also a portal user, access to user content stored through the portal is possible from a scenario. For more information about Content Server see Chapter 9, "Managing Content Repositories." For more information about configuring the CMIS provider, see Section 25.4.2, "Configuring the CMIS Provider."

Activity Graph Data Provider

The Activity Graph data provider is an out-of-the-box provider that you can optionally use as a data source in your Personalization project. Activity stream information from a portal or Portal Framework application is exposed through Activity Graph. The Activity Graph REST service runs on the `WC_Spaces` server and provides access to activity stream based recommendations as formed by the activity graph.

If a Personalization user is also a portal user, access to activity related recommendations (for portal content-types) is possible from a scenario. For more information about the Activity Graph service, see Chapter 10, "Managing Activity Graph." For more information about configuring the Activity Graph provider, see Section 25.4.3, "Configuring the Activity Graph Provider."

Oracle People Connections Locator

The People Connections locator is a locator that you can optionally use as a data source in your Personalization project. People Connections information is exposed through the People Web service. The People Connection REST service runs on the `WC_Spaces` server and provides access to social profile data as created in the context of a portal. If a Personalization user is also a portal user, access to People profile data is possible from a scenario. For more information about People Connections, see the "Integrating People Connections" chapter in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

Unlike the other out-of-the-box data providers, the People Connection Web service is accessed through the general purpose Property Service data provider using the IPropertyLocator extension interface. For more information about configuring the People Connections provider, see Section 25.4.4, "Configuring the Oracle People Connections Locator."

25.3 Configuring the WebCenter OPSS Trust Service

Personalization leverages a new feature from OPSS (Oracle Platform Security Services) for single-sign-on. Enabling this feature by following the configuration steps described here is required in all but the simplest Personalization use cases.

The OPSS Trust Service does not need to be configured when:

- Directly interacting with the Conductor and Property service from a REST client
- The Conductor and Property Service are being used by Personalization client libraries from a custom JEE Web application deployed in the same domain as Personalization, if JSessionId has been configured for both Web applications (note that there will be many exceptions logged making debugging difficult)

The OPSS Trust Service must be configured when:

- Personalization is part of a production deployment
- There are Personalization scenarios that require the out-of-the-box data providers (Activity Graph, CMIS, and People Connections Locator)
- The Conductor and Property Service are being used by Personalization client libraries from a custom JEE Web application deployed in the same domain as Personalization
- Cross-domain trust is required (i.e., an integrated domain connection configured to use the `WC_domain` CMIS provider)

This section contains the following subsections:

- Section 25.3.1, "Configuring the Trust Service in the Oracle WebCenter Portal Domain"
- Section 25.3.2, "Configuring the Trust Service in the Integrated WLS Domain"
- Section 25.3.3, "Configuring Cross-Domain Trust"

25.3.1 Configuring the Trust Service in the Oracle WebCenter Portal Domain

The default Oracle WebCenter Portal installation includes the Personalization domain extension template, which installs two WLST python scripts (`configureTrustWCPS.py` and `configureConnectionsWCPS.py`), in the domain home:

```
oracle/user_projects/applications/wc_domain/scripts
```

These scripts and associated `configureWCPS.properties` file contain usage instructions. Note that these are sample scripts, and that before running the scripts, you must edit the properties file and, at a minimum, specify the `ocs.server` name (typically the Content Server), the `spaces.server.host` name, and the `fmwconfig.location`. These values are unique to each Oracle WebCenter Portal installation and must be edited. Other values may also need to be changed according to the local environment (the machine port numbers, for example, may be different).

The `configureConnectionsWCPS.py` script sets up the default Personalization connection information for you (i.e., connection information for Activity Graph, CMIS, and People Connections). The script relies on the `WCPS.py` library, which is only installed on the Oracle WebCenter Portal domain (and not in the integrated WLS domain). You can, however, run `configureConnectionsWCPS.py` in the Oracle WebCenter Portal domain and point it (using a t3 URL) to an integrated WLS domain.

Caution: the Trust Service configuration set up by `configureTrustWCPS.py` should not be applied remotely. The script should only be run from the WebCenter Portal domain (`wc_domain`).

You must use the `WCP_ORACLE_HOME/common/bin/wlst.sh` command file (for example, `$WCP_ORACLE_HOME/common/bin/wlst.sh configureconnectionsWCPS.py`) that sets up environment variables correctly for these scripts.

After running the scripts, restart all servers in the domain including the Admin server.

Testing the Configuration

To see Trust Service single sign-on in operation, you must be calling the Conductor or Property Service from a custom JEE Web application (using the Personalization client libraries), or be executing a scenario that uses a Personalization connection (such as the Activity Graph or CMIS data providers), or accessing a People Connections property using the People Connections locator.

When doing any of the above, you should see the following default log entry in `WC_Utilities-diagnostic.log`:

```
[2010-11-10T07:30:40.362-08:00] [WC_Utilities] [NOTIFICATION] []
[oracle.jps.trust] [tid: [ACTIVE].ExecuteThread: '3' for queue:
'weblogic.kernel.Default (self-tuning)']
[ecid: 0000IkqQG4NBh49LJeCCyf1CqfXw000008,0] [APP: wcps-services#11.1.1.4.0] Token
issue operation
```

You should also see the following default log entry in the `WC_Spaces-diagnostic.log` (if accessing services there):

```
[2010-11-10T07:30:40.236-08:00] [WC_Spaces] [NOTIFICATION] [] [oracle.jps.trust]
[tid: [ACTIVE].ExecuteThread: '1' for queue: 'weblogic.kernel.Default
(self-tuning)']
[ecid: d461d36d4a552b90:-1fe62a5d:12c365bb19b:-8000-000000000000002c,0] [APP:
webcenter#11.1.1.4.0] Token validate operation.
```

25.3.2 Configuring the Trust Service in the Integrated WLS Domain

A separate python script is shipped with the JDeveloper installer to configure the integrated WLS domain located in the following directory:

```
DefaultDomain\scrpts-wcps\
```

This script can be run manually or using JDeveloper's **External Tools** function.

Edit the properties file if you are using a non-default user or password. After creating and starting the integrated WLS domain, run the script from the `scrpts-wcps` directory:

```
Oracle\MiddlewareRC8\oracle_common\common\bin\wlst.cmd configureWCPS.py
configureWCPS.properties
```

Note: The `configureTrustWCPS.py` script is called when you select **Configure WCPS Trust Service** from the Tools menu. By default, this script refers to port 7101 on the Integrated WebLogic Server. If this default Integrated WebLogic Server port number was changed for some reason, you must update the `configureWCPS.properties` file with the correct port number. The script and properties file are located in the domain, here:

`\system*\DefaultDomain\scrpt-wcps\configureWCPS.properties`

`\system*\DefaultDomain\scrpt-wcps\configureWCPS.py`

You need to change this line in the configureWCPS.properties file:

`admin.url=t3://localhost:7101`

Restart the integrated WLS domain.

Testing the Configuration

Default logging levels are not enough to confirm token-issue and token-validate operations. Use the **Configure Oracle Diagnostic Logging** feature in JDeveloper and navigate to the `oracle.jps.trust` logger and set the level to `Finest` as shown below:

1. From the View menu, select **ApplicationServer Navigator**.
2. Expand **Application Servers**.
3. Right-click **IntegratedWeblogicServer** and select **Configure Oracle Diagnostic Logging**.
4. Set the logging level to `Finest`

Now run a scenario involving a custom JEE Web application calling the Conductor or Property Services.

25.3.3 Configuring Cross-Domain Trust

The Trust Service supports cross-domain trust, meaning if keystores have been created in different WLS domains, a client may allocate a token in domain 'A', issue an HTTP request with the token to domain 'B', and have the identity asserter validate and authenticate the user/request in domain 'B' through single sign-on. Note that a key assumption is that the user in domain 'A' exists in, and is the same user in domain 'B'.

By default, when running the `configureWCPS.py` script in the integrated WLS domain a certificate named `extDomain.cer` is generated. To enable cross-domain trust between the integrated WLS domain and Oracle WebCenter Portal domain:

Copy `extDomain.cer` to your Oracle WebCenter Portal domain (`wc_domain`) installation and import it there. Copy the `extDomain.cer` file to the scripts location:

```
oracle/user_projects/applications/wc_domain/scripts
```

Type in the following command to import the certificate:

```
keytool -importcert -alias orakey1 -file extDomain.cer -keystore
../../../../wlserver_10.3/wc_domain/config/fmwconfig/default-keystore.jks
-storepass weblogic
```

Restart the servers in the Oracle WebCenter Portal domain.

Testing the Configuration

The simplest way to validate cross domain trust is to create a People Connections Personalization connection in the integrated WLS domain that points to the WebCenter Portal domain's `WC_Spaces` server. Then, create and deploy a simple scenario to the integrated WLS domain that fetches a People Connections property value. Finally, confirm that the 'Token Validate Operation' message described above is logged on the `WC_Spaces` server.

25.4 Configuring Providers

Personalization for WebCenter Portal provides out-of-the-box providers for Activity Graph and the Content Server, and a locator for People Connections. For scenarios using any of these providers, you must configure them using the `configureWCPS.py` WLST script as described in the following sections. If you are using custom providers or locators, then you must also configure them as described in the section on

configuring custom providers. You do not need to configure providers or locators if they are not being used in your scenarios.

You can develop scenarios without the out-of-the-box providers, or exclusively with custom providers or downloaded from the Oracle Technology Network (OTN). Also, if you are developing exclusively within the JDeveloper integrated domain, you do not ordinarily have access to these `WC_Spaces`-based services (since WebCenter Portal does not run in the integrated domain). With advanced configurations (also supported by `configureWCPS.py`) you can access the WebCenter Portal services in the `WC_Spaces` domain from the integrated domain's Personalization server. This uses cross-domain trust and does require the provider connections to be configured.

The `configureTrustWCPS.py` and `configureConnectionsWCPS.py` scripts (located in the `WC_Spaces` domain), or `configureWCPS.py` for JDeveloper's integrated WLS domain (located in the `DefaultDomain/scrpts-wcps` domain directory) are used to configure the corresponding domains by pointing to the appropriate WLS Administration server.

- Section 25.4.1, "Creating or Modifying Provider Connection Settings"
- Section 25.4.2, "Configuring the CMIS Provider"
- Section 25.4.3, "Configuring the Activity Graph Provider"
- Section 25.4.4, "Configuring the Oracle People Connections Locator"
- Section 25.4.5, "Configuring Custom Providers"

25.4.1 Creating or Modifying Provider Connection Settings

This section describes how to use WLST and Fusion Middleware Control to create or change the connection information (stored in `wcps-connections.xml`). It also describes how you can write a custom configuration class to configure a custom provider.

This section contains the following subsections:

- Section 25.4.1.1, "Understanding Personalization Connection Information"
- Section 25.4.1.2, "Connection Configuration Attributes"
- Section 25.4.1.3, "Configuring Connections Using WLST"
- Section 25.4.1.4, "Configuring Connections Using Fusion Middleware Control"
- Section 25.4.1.5, "Writing a Custom Configuration Class"

25.4.1.1 Understanding Personalization Connection Information

Personalization connection information is maintained in `wcps-connections.xml`, which can be found in the domain directory at the following location:

```
<domain directory>/config/fmwconfig/wcps-connections.xml
```

Although editing this file directly is not recommended, there are several ways in which you can modify connection information:

- WLST - you can write a script with WLST commands to access the system MBeans representing the connection configuration. For more information on using WLST commands to configure connection settings, see Section 25.4.1.3, "Configuring Connections Using WLST."
- Fusion Middleware Control - you can use Fusion Middleware Control to view or edit the JMX MBeans deployed with Personalization. For more information on

using Fusion Middleware Control to configure connection settings, see Section 25.4.1.4, "Configuring Connections Using Fusion Middleware Control."

25.4.1.2 Connection Configuration Attributes

Connection properties are maintained in `wcps-connections.xml` in the following form:

```
<properties>
    <property>
        <name>{property name}</name>
        <value>{property val}</value>
    </property>
```

The following shows the connection properties and attributes that can be modified using WLST or Fusion Middleware Control. Note that each connection property is specific to the provider or locator that the connection is for. For example, the CMIS provider will have different connection properties than the Activity Graph provider.

- **`<connection-name>`** - unique name for this connection. Connections can be retrieved by name.
- **`<connection-type>`** - unique type for this connection. Connections can be retrieved by type. Note that `connection-type` only needs to be specified for custom connections. For the out-of-the-box data providers this field is set internally.
- **`<namespace>`** - generally, this must match the namespace the accessing scenario is deployed within. The namespace is how the Conductor determines how to locate the appropriate <connection> for a given scenario. You can use a wildcard '*' to make this connection element available in all namespaces. If left unspecified in the WLST script, `namespace` will default to '*'.
- **`<name>isDefault</name>`** - marks this connection as the default connection (if multiple are defined). Since it is a connection property it must have the form:

  ```
  <properties>
              <property>
                  <name>isDefault</name>
                  <value>true</value>
              </property>
  </properties>
  ```

  Note that multiple connections can have the "isDefault" flag set to true. If this is the case, it is up to the individual provider to return the default connection.

25.4.1.3 Configuring Connections Using WLST

A set of Personalization WLST commands is provided to allow easy configuration of your provider connections. You can combine these commands into a script, an example of which is provided that can be customized or used directly. The sample script sets up provider connections and also initializes the Trust Services.

The Personalization WLST commands are installed at `WCP_ORACLE_HOME/common/wlst/WCPS.py` and are invoked using the `WCP_ORACLE_HOME/common/bin/wlst.sh`(cmd) script.

Each out-of-the-box data provider is supported with specific WLST commands (described in sections below). For custom data providers, use the generic WLST commands to configure a connection. For example:

```
createWCPSCustomConnection('customConnectionName',
'connectionType', properties={ 'name1': 'value1', 'name2':
'value2' })
```

For a complete list of Personalization WLST connection and other commands, see the "Personalization" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

25.4.1.4 Configuring Connections Using Fusion Middleware Control

You can use Fusion Middleware Control to view or edit the JMX MBeans (the connection configuration MBeans for Activity Graph, Content Server, and People Connections deployed with Personalization).

To view or edit connection configuration MBeans:

1. Open Fusion Middleware Control Navigate to **Personalization Services**.
2. Click **WCPS-Services**.
3. From the Application Development drop-down menu, select **System MBean Browser**.
4. In the MBean browser under 'Application Defined MBeans', select `oracle.wcps.connections` and continue to drill down to the connection information you wish to modify.
5. On the Attributes tab, select **Properties** to view current values of connection attributes.
6. On the Operations tab, select **setProperty** and click **Invoke** to modify the name/value pairs.

25.4.1.5 Writing a Custom Configuration Class

Custom configuration classes (classes annotated with `@ConnectionConfiguration`) are implemented by customers writing their own data providers. This allows custom data providers to use the Personalization connection framework to retrieve connection information configured using the Personalization WLST scripts.

Custom configuration classes for data providers are more fully described in the "Implementing Custom Data Providers: Introduction" chapter in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

25.4.2 Configuring the CMIS Provider

If you are working outside of the Integrated WLS domain (i.e., in the `WC_Spaces` domain), before you can use the CMIS provider in your scenario, you must first configure connection settings for it.

Connection settings for the CMIS provider are maintained in `wcps-connections.xml` in the `wc_domain` conductor extensions directory (domain scoped). Although it is not recommended that you edit this file directly, there are several other ways in which you can modify connection settings. Use one of the methods described in Section 25.4.1, "Creating or Modifying Provider Connection Settings." to configure the connection settings for the CMIS provider.

Connection Element and Property Descriptions

`<connection-name>` - unique name for this connection. Connections can be retrieved by name.

<connection-type>cmis.provider.connection</connection-type> - defines the connection type for the CMIS provider.

<namespace> - generally, this must match the namespace the accessing scenario is deployed within. The namespace is how the Conductor determines how to locate the appropriate `<connection>` for a given scenario. Using a wildcard '*', you can make this connection element available in all namespaces.

<name>isDefault</name> - marks this connection as the default connection (if multiple connections are defined)

<name>repositoryId</name> - this property must be changed to match the Content Repository in your environment.

<name>path</name> - this property must be changed to match the Content Repository in your environment. Defaults to `/api/cmis/repository/repositoryId` if not specified

<name>scheme</name> - protocol to access the CMIS REST service (HTTP or HTTPS for SSL). Defaults to `HTTP` if not explicitly specified.

<name>host</name> - machine name hosting the CMIS REST service. This is the machine name of the `WC_Spaces` managed server.

<name>port</name> - machine port number hosting the CMIS REST service. This is the machine port number of the WC_Spaces managed server.

<name>pathTrim</name> - Default is no trim if not explicitly specified

<name>rewriteUrls</name> - can be set to `None`, `Consumer`, or `Producer`. See the REST Proxy page for details. If you want the direct URLs (from the CMIS server for document links), set this to `None`. Default is no rewrite (`None`).

<name>username</name> - (Optional) the username to use when connecting to the CMIS REST service. Can be used to force a connection to a fixed username.

<name>password</name> - (Optional) the password to use when connecting to the CMIS REST service. Used in conjunction with `username`, can be used to force a connection to a fixed username/password. Not recommended for use outside of development environment since password access is not secured.

<name>timeoutInMillisecs</name> - Time in milliseconds before the CMIS query read response times out

<name>propagateTimeoutExceptions</name> - If true, propagate the timeout exceptions. Otherwise, log the exception and return null for the CMIS query response.

<name>restResourceURL</name> - the restResourceURL is the REST URL endpoint that defines the API resources for the CMIS REST service.

```
<property>
    <name>restResourceURL</name>
    <value><a target="_blank"
href="http://my.webcenter.host:8888/rest/api/resourceIndex">http://my.webcenter.ho
st:8888/rest/api/resourceIndex</a></value>
</property>
```

25.4.3 Configuring the Activity Graph Provider

If you are working outside of the Integrated WLS domain (i.e., in the `WC_Spaces` domain), before you can use scenarios that rely on the Activity Graph provider, you must configure connection information for your local environment.

Connection settings for Activity Graph are maintained in `wcps-connections.xml` in the `wc_domain` conductor extensions directory (domain scoped). Although it is not recommended that you edit this file directly, there are several other ways in which you can modify connection settings. Use one of the methods described in Section 25.4.1, "Creating or Modifying Provider Connection Settings" to configure the connection settings for the Activity Graph provider.

Connection Element and Property Descriptions

`<connection-name>` - unique name for this connection. Connections can be retrieved by name.

`<connection-type>activity.provider.connection</connection-type>` - defines the connection type for the Activity Graph provider.

`<namespace>` - generally, this must match the namespace the accessing scenario is deployed within. The namespace is how the Conductor determines how to locate the appropriate `<connection>` for a given scenario. Using a wildcard '*', you can make this connection element available in all namespaces.

`<name>isDefault</name>` - marks this connection as the default connection (if multiple connections are defined)

`<name>scheme</name>` - protocol to access the Activity Graph REST service (HTTP or HTTPS for SSL). Defaults to `HTTP` if not explicitly specified.

`<name>host</name>` - machine name hosting the Activity Graph REST service. This is the machine name of the `WC_Spaces` managed server.

`<name>port</name>` - machine port number hosting the Activity Graph REST service. This is the machine port number of the `WC_Spaces` managed server.

`<name>rewriteUrls</name>` - can be set to `None`, `Consumer`, or `Producer`. See the REST Proxy page for details. If you want the direct URLs (from the CMIS server for document links), set this to `None`. Default is no rewrite (`None`).

`<name>user</name>` - (Optional) the username to use when connecting to the Activity Graph REST service.

`<name>password</name>` - (Optional) the password to use when connecting to the Activity Graph REST service. Used in conjunction with `username`, can be used to force a connection to a fixed username/password. Not recommended for use outside of development environment since password access is not secured.

`<name>restResourceIndex</name>` - the URI suffix to append to the host/port for the REST resource index (for example: '/rest/api/resourceIndex')

`<name>restResourceURL</name>` - the restResourceURL is the REST URL endpoint that defines the API resources for the Activity Graph REST service.

```
<property>
   <name>restResourceURL</name>
   <value><a target="_blank"
href="http://my.webcenter.host:8888/rest/api/resourceIndex">http://my.webcenter.ho
st:8888/rest/api/resourceIndex</a></value>
</property>
```

25.4.4 Configuring the Oracle People Connections Locator

If you are working outside of the Integrated WLS domain (i.e., in the `WC_Spaces` domain), before you can use scenarios that rely on the People Connections locator, you must configure connection information for your local environment.

The Property Service uses an IPropertyLocator (the People Connections IPropertyLocator) to interface with the People Connections service. The Property Provider that interfaces with the People Connections service uses this IPropertyLocator to make People Connections REST calls for a given user (or self) and return the 'Person' object represented by that REST service call. The Person attributes represent values for that WebCenter Portal profile.

Connection settings for the People Connections IPropertyLocator are maintained in `wcps-connections.xml` in the `wc_domain` conductor extensions directory (domain scoped). Although it is not recommended that you edit this file directly, there are several other ways in which you can modify connection settings. Use one of the methods described in Section 25.4.1, "Creating or Modifying Provider Connection Settings." to configure the connection settings for the People Connections IPropertyLocator.

Connection Element and Property Descriptions

`<connection-name>` - unique name for this connection. Connections can be retrieved by name.

`<namespace>` - generally, this must match the namespace the accessing scenario is deployed within. The namespace is how the Conductor determines how to locate the appropriate <connection> for a given scenario. Use a wildcard '*' to make this connection element available in all namespaces.

`<name>isDefault</name>` - marks this connection as the default connection (if multiple are defined)

`<name>scheme</name>` - protocol to access the People Connections REST service (HTTP or HTTPS for SSL). Defaults to `HTTP` if not explicitly specified.

`<name>host</name>` - machine name hosting the People Connections REST service. This is the machine name of the `WC_Spaces` managed server.

`<name>port</name>` - machine port number hosting the People Connections REST service. This is the machine port number of the `WC_Spaces` managed server.

`<name>user</name>` - (Optional) the username to use when connecting to the People Connections REST service. Can be used to force a connection to a fixed username.

`<name>password</name>` - (Optional) the password to use when connecting to the People Connections REST service. Used in conjunction with `username`, can be used to force a connection to a fixed username/password. Not recommended for use outside of development environment since password access is not secured.

`<name>restResourceIndex</name>` - appended to the People Connections REST service host/port, pointing to the location of the `resourceIndex` (available REST services) page (for example: '`/rest/api/resourceIndex`')

Bootstrapping the Person class to the Properties Provider

In order for the Property Service to know about and use the results of the People Connections REST calls, it needs to know about a 'Person'. This means creating a 'Person' property set definition, along with its individual attributes set as property definitions, before a 'Person' can be instantiated and its properties set.

The People Connections IPropertyLocator code does this by bootstrapping that process in a servlet listener, configured in its `web.xml` file as shown in the example below:

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

```
<web-app xmlns="http://java.sun.com/xml/ns/j2ee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
http://java.sun.com/xml/ns/j2ee/web-app_2_4.xsd"
    version="2.4">

    <!-- Allows connection config classes to be loaded from the connections.xml
file -->
     <listener>

<listener-class>oracle.wcps.connection.lifecycle.VersatileModuleLifecycleCallback<
/listener-class>
    </listener>

     <listener>

<listener-class>oracle.wcps.people.util.BootstrapPropertiesListener</listener-clas
s>
    </listener>

</web-app>
```

Accessing the People Connections REST connection configuration information

Because the People Connections IPropertyLocator is not an actual 'provider', you cannot access its connection information in provider code, as for example Activity Graph. Instead, internal code uses the `VersatileModuleLifecycleCallback` class to make that configuration information available, and consequently makes this is a required listener to be configured in `web.xml`. Once that listener has been activated, the code can make calls to access connection information and parameterize the IPropertyLocator code to point to the People Connections REST server.

Tying the PC IPropertyLocator to the Properties Service Provider

The Properties Service Provider (PSP) knows about the People Connections IPropertyLocator through the namespaces. In the bootstrapping process above, the `PropertySetDefinition` (and its definitions) are namespaced with "`people.connections.person`" and a locator class "`oracle.wcps.people.property.PeoplePropertyLocator`". In using the PSP, the `namespace` and `PropertySetDefinition` are passed to the PSP. The PSP uses the locator class defined in the `PropertySetDefinition` to instantiate and delegate to property values (Person values from the People Connections REST service).

The locator and namespaces are critical in this process. They are defined as constants in the internal PropertyDefConstants class. Note that these do not need to be configured.

25.4.5 Configuring Custom Providers

Connection settings for Activity Graph are maintained in `wcps-connections.xml` in the `wc_domain` conductor extensions directory (domain scoped). Although it is not recommended that you edit this file directly, there are several other ways in which you can modify connection settings. Use one of the methods described in Section 25.4.1, "Creating or Modifying Provider Connection Settings." to configure the connection settings for your custom provider. Refer to Section 25.4.1.2, "Connection Configuration Attributes" for information about the configuration file elements and property descriptions.

25.5 Configuring Coherence

If your installation is using Coherence for caching (a requirement for "high-availability" environments), four separate caches must be set up: one each for Namespaces, Property Definitions, Property Set Definitions, and Property Sets.

The sample `wcps-cache-config.xml` configuration file below shows how to configure simple Coherence local caches. For more advanced cache types, refer to the Coherence documentation. Note that Coherence classes must be accessible via the same class loader as Personalization. The Coherence `wcps-cache-config.xml` file must also be accessible by that same class loader. For more information, see the `oracle.wcps.cache.CacheFactory` class in the JavaDoc for WebCenter Portal in *Oracle Fusion Middleware Java API Reference for Oracle WebCenter Portal*. See also the "Configuring Coherence as the Caching Mechanism" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

```
<?xml version="1.0" encoding="UTF-8" ?>
<cache-config>
    <caching-scheme-mapping>
        <cache-mapping>
            <cache-name>com.oracle.p13n.service.property.namespaces</cache-name>
            <scheme-name>ns-local-side</scheme-name>
        </cache-mapping>
        <cache-mapping>

<cache-name>com.oracle.p13n.service.property.propertydefinitions_*</cache-name>
            <scheme-name>pd-local-side</scheme-name>
        </cache-mapping>
        <cache-mapping>

<cache-name>com.oracle.p13n.service.property.propertysetdefinitions_*</cache-name>
            <scheme-name>psd-local-side</scheme-name>
        </cache-mapping>
        <cache-mapping>

<cache-name>com.oracle.p13n.service.property.propertysets_*</cache-name>
            <scheme-name>ps-local-side</scheme-name>
        </cache-mapping>
        <cache-mapping>
            <cache-name>*</cache-name>
            <scheme-name>properties-default-local</scheme-name>
        </cache-mapping>
    </caching-scheme-mapping>

    <caching-schemes>
<!--
The following schemes are all local.  For a clustered deployment,
a distributed, replcated, or other clustered scheme is recommended.
See Coherence documentation for more information.
-->

        <local-scheme>
            <scheme-name>ns-local-side</scheme-name>
            <service-name>NamespaceCache</service-name>

            <eviction-policy>HYBRID</eviction-policy>
            <high-units>{back-size-limit 0}</high-units>
            <unit-calculator>FIXED</unit-calculator>
            <expiry-delay>{back-expiry 1h}</expiry-delay>
            <flush-delay>1m</flush-delay>
```

```
            <cachestore-scheme/>
        </local-scheme>

        <local-scheme>
            <scheme-name>pd-local-side</scheme-name>
            <service-name>PropertyDefinitionCache</service-name>

            <eviction-policy>HYBRID</eviction-policy>
            <high-units>{back-size-limit 0}</high-units>
            <unit-calculator>FIXED</unit-calculator>
            <expiry-delay>{back-expiry 1h}</expiry-delay>
            <flush-delay>1m</flush-delay>

            <cachestore-scheme/>
        </local-scheme>

        <local-scheme>
            <scheme-name>psd-local-side</scheme-name>
            <service-name>PropertySetDefinitionCache</service-name>

            <eviction-policy>HYBRID</eviction-policy>
            <high-units>{back-size-limit 0}</high-units>
            <unit-calculator>FIXED</unit-calculator>
            <expiry-delay>{back-expiry 1h}</expiry-delay>
            <flush-delay>1m</flush-delay>

            <cachestore-scheme/>
        </local-scheme>

        <local-scheme>
            <scheme-name>ps-local-side</scheme-name>
            <service-name>PropertySetCache</service-name>

            <eviction-policy>HYBRID</eviction-policy>
            <high-units>{back-size-limit 0}</high-units>
            <unit-calculator>FIXED</unit-calculator>
            <expiry-delay>{back-expiry 1h}</expiry-delay>
            <flush-delay>1m</flush-delay>

            <cachestore-scheme/>
        </local-scheme>

    <local-scheme>
            <scheme-name>properties-default-local</scheme-name>
            <service-name>DefaultCache</service-name>

            <eviction-policy>HYBRID</eviction-policy>
            <high-units>{back-size-limit 0}</high-units>
            <unit-calculator>FIXED</unit-calculator>
            <expiry-delay>{back-expiry 1h}</expiry-delay>
            <flush-delay>1m</flush-delay>

            <cachestore-scheme/>
        </local-scheme>

    </caching-schemes>

</cache-config>
```

25.6 Configuring Content Presenter

Before you can run Personalization scenarios using Content Presenter, you need to configure the connections file (`connections.xml`) so that Content Presenter can see your Conductor server and the tagged scenarios. For more information about Content Presenter, see the "Publishing Content Using Content Presenter" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

The `connections.xml` file holds the connection information for the application that you're working with on the WebCenter Portal side. Content Presenter gets a list of all URL connections that are registered within this file and for any that begin with "`Conductor`", Content Presenter will assume this is a URL pointing to a Conductor server. For more information about `connections.xml`, see Appendix A, "WebCenter Portal Files" in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

Because Content Presenter runs on the `WC_Spaces` server while Personalization scenarios are executed on the `WC_Utilities` server, you also need to configure the cross-domain trust service. For more information, see Section 25.3, "Configuring the WebCenter OPSS Trust Service."

You can configure the Content Presenter task flow either at runtime, or from JDeveloper when adding the Content Presenter task flow to a page. These two configuration options for Content Presenter, as well as configuration requirements on the Conductor side are described in the following subsections:

- Section 25.6.1, "Configuring the WebCenter Portal Application's Content Server Connection"
- Section 25.6.2, "Configuring the Content Presenter Task Flow Parameters"
- Section 25.6.3, "Configuring the Conductor's Scenario Tags"

25.6.1 Configuring the WebCenter Portal Application's Content Server Connection

For a Portal Framework application, use JDeveloper to set up the URL connection, or set the Content Presenter task flow parameters as described in Section 25.6.2, "Configuring the Content Presenter Task Flow Parameters." For WebCenter Portal, you can use either WLST commands or Fusion Middleware Control to configure the connection.

This section contains the following subsections:

- Section 25.6.1.1, "Configuring Connections for WebCenter Portal Using WLST"
- Section 25.6.1.2, "Configuring Connections for WebCenter Portal Using Fusion Middleware Control"

25.6.1.1 Configuring Connections for WebCenter Portal Using WLST

For WebCenter Portal, you can use the `adf_createHttpUrlConnection` WLST command to manage URL connections as shown in the following example:

```
adf_createHttpUrlConnection('webcenter','Conductor','http://example.com:8891/wcps/
api/conductor?namespace=CP_namespace&
```

For more information about the `adf_createHttpUrlConnection` command, see the "adf_createHttpUrlConnection" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

25.6.1.2 Configuring Connections for WebCenter Portal Using Fusion Middleware Control

For WebCenter Portal, you can use Fusion Middleware Control to configure connections.

To configure connections using Fusion Middleware Control:

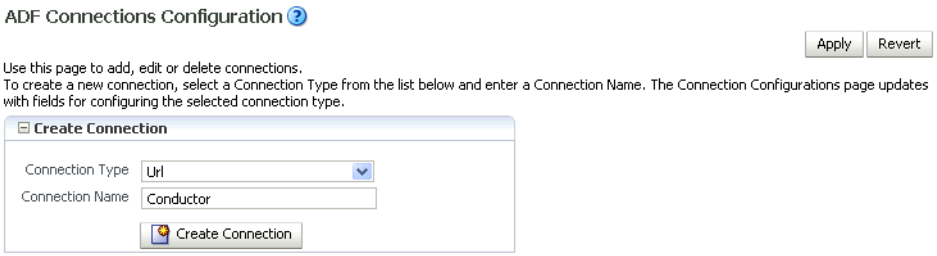
1. Log in to Fusion Middleware Control and navigate to the home page for the WebCenter Portal application.

   See: Section 6.2, "Navigating to the Home Page for WebCenter Portal"

2. From the **WebCenter Portal** menu, select **ADF -> Configure ADF Connections**.

   The ADF Connections Configuration page displays (see Figure 25–1).

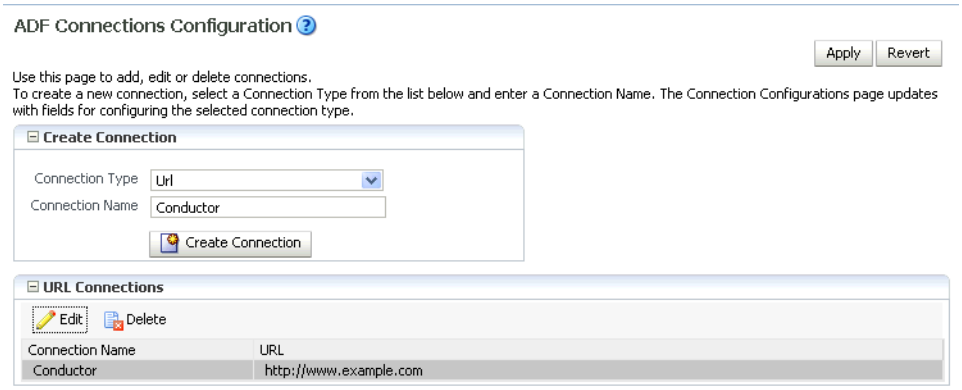
Figure 25–1 ADF Connections Configuration Page



3. Set the **Connection Type** to `Url`, enter the **Connection Name** as `Conductor`, and click **Create Connection**.

   The URL Connections section displays (see Figure 25–2).

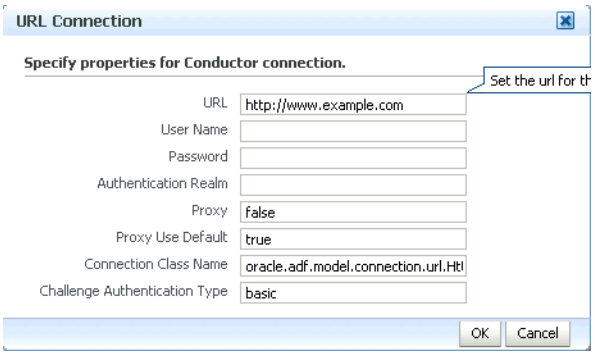
Figure 25–2 ADF Connections Configuration Page (URL Connections)



4. Click the **Edit** (Pencil) icon.

   The URL Connection dialog displays (see Figure 25–3).

Figure 25–3 URL Connection Dialog



5. Edit the URL so that it points to the Content Server instance used by the WebCenter Portal application. For example:

   ```
   http://example.com:8891/wcps/api/conductor?namespace=CP_namespace&repoId=myhost-ucm11g
   ```

6. Update any other fields as required for your local connection and click **OK**.

25.6.2 Configuring the Content Presenter Task Flow Parameters

You can configure a Content Presenter instance through its task flow parameters. These can either be set at runtime in the Content Presenter Configuration dialog, or from JDeveloper as you add the Content Presenter task flow. To do this manually, you need to set two parameters: the `Data Source Type` parameter must be set to `dsTypeScenarioResults`, and the `Data Source` parameter should be set to something like:

```
conductor-connection-name=ConductorConnectionName,namespace=ScenarioNamespace,scenario-name=ScenarioName,inputparam1=value1,inputparam2=value2
```

Note that the `conductor-connection-name` value must match with a URL connection that points to a valid Conductor server. Also, the namespace used should be the name of the namespace to which the specified scenario belongs. Finally, any input parameters that the scenario may be expecting can appended (as shown above), with a comma separating the name/value pairs.

For more information about configuring Content Presenter for Personalization, see:

- At runtime: "Setting Content Presenter Task Flow Properties" in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.
- In JDeveloper: "Content Presenter Task Flow Parameters and Out-of-the-Box Display Templates" in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

25.6.3 Configuring the Conductor's Scenario Tags

On the Conductor side, your Scenarios must have the correct tag in order for Content Presenter to see them. Content Presenter uses the tag `ContentPresenterScenario`, so any scenarios you want Content Presenter to pick up must have this tag associated with them.

Once you have everything set up on both the Conductor and WebCenter Portal application side, everything else is fairly simple. When you open the Content

Presenter Configuration dialog at runtime, the Content Source selection list displays **Results of a Scenario**. Selecting this displays a table of all of the scenarios that have been tagged for Content Presenter consumption. The first Scenario is always selected, and if it has any Input Parameters defined, they will be displayed below the table with input fields.

As you select scenarios in the table, the Input Parameters below will be updated. After selecting a scenario and specifying any input parameters, you can either preview or save Content Presenter to get the results. The results will be displayed much like any other multi-valued content source and will ultimately depend on the template selected for display purposes.

Note: Any results that are returned from a scenario for use within Content Presenter must return a valid CMIS query as Content Presenter takes the return value and runs it (as a CMIS query) against the repository specified within the Conductor URL.

25.7 Configuring Single Sign-On

Single sign-on is configured as part of running the `configureTrustWCPS.py` and `configureConnectionsWCPS.py` scripts for configuring the Oracle WebCenter Portal domain, or the `configureWCPS.py` script for configuring JDeveloper's integrated WLS domain. When you run these scripts they also set up Trust Services single sign-on, which allows single sign-on for REST HTTP requests between client JEE Web applications, the Personalization Web application, and the `WC_Spaces` Web application REST services used by the out-of-the-box data providers. All these applications are also configured to support OAM/OSSO-provided single sign-on tokens, as well, without any additional Personalization-specific configuration. For more information, see Section 25.3.3, "Configuring Cross-Domain Trust."

25.8 Overriding the Default Security Settings

By default, all access to Personalization REST resources (other than the resourceIndex) requires authentication. In most cases this will be sufficient for development. However, for production environments, you may want to modify the default security constraints. The following sections describe how to set up less security to execute scenarios (where anonymous access is needed), and more security to prevent the ability to create new scenarios.

This section contains the following subsections:

25.8.1 Allowing Anonymous Execution of Scenarios

Adding the following security constraint to the domain's `conductor-extensions-library\WEB-INF\web.xml`, file will honor default descriptor (authentication required) security, plus allow anonymous GET/POST on scenarios created or deployed from an anonymous application or namespace:

```
<security-constraint>
  <web-resource-collection>
    <web-resource-name>ConductorJerseyWebApplication</web-resource-name>
```

```
    <url-pattern>/api/conductor/namespaces/anonymous/scenarios/*</url-pattern>
    <http-method>GET</http-method>
    <http-method>POST</http-method>
  </web-resource-collection>
  <!-- Allow anonymous user access  in this namespace -->
  <!--auth-constraint>
     <role-name>Users</role-name>
   </auth-constraint-->
</security-constraint>
```

25.8.2 Disabling Scenario Creation by Anonymous Users

You can disable scenario creation for any anonymous user using a simple filter that you add to the domain's `conductor-extensions-library\WEB-INF\web.xml` file. An example filter (AnonymousScenarioFilter) is shown below:

```
<filter>
    <filter-name>AnonymousScenarioFilter</filter-name>
    <filter-class>oracle.wcps.samples.AnonymousScenarioFilter</filter-class>
  </filter>
  <filter-mapping>
    <filter-name>AnonymousScenarioFilter</filter-name>

<url-pattern>/api/conductor/namespaces/PublicPersonalization/scenarios</url-patter
n>
  </filter-mapping>
  <security-constraint>
      <web-resource-collection>
          <web-resource-name>ConductorJerseyWebApplication</web-resource-name>

<url-pattern>/api/conductor/namespaces/PublicPersonalization/scenarios/*</url-patt
ern>
          <http-method>GET</http-method>
          <http-method>POST</http-method>
      </web-resource-collection>
      <!--
        Commented out to allow anonymous user access  in this namespace
      <auth-constraint>
      <role-name>Users</role-name>
      </auth-constraint>
      -->
  </security-constraint>
```

The filter explicitly checks for an anonymous user (`httpRequest.getUserPrincipal() == null`) and an HTTP `POST` operation on the `/api/conductor/namespaces/PublicPersonalization/scenarios` URL, which is the REST resource for creating a scenario.

```
public void doFilter(ServletRequest request, ServletResponse response, FilterChain
chain)
        throws IOException, ServletException
    {
        if (request instanceof HttpServletRequest)
        {
            HttpServletRequest httpRequest = (HttpServletRequest)request;
            String method = httpRequest.getMethod();
            if ("POST".equalsIgnoreCase(method) && httpRequest.getUserPrincipal()
== null)
            {
              throw new ServletException("Anonymous users cannot create
```

```
scenarios.");
            }
        }
        chain.doFilter(request, response);
    }
```

25.8.3 Disabling Scenario Creation by Authenticated Users

A simple change to the filter described in Section 25.8.2, "Disabling Scenario Creation by Anonymous Users" (removing the `httpRequest.getUserPrincipal()` check) would disable scenario creation for all users. Although the HTTP POST operation is also used to request execution of scenarios, the URI in that case is different (and protected in the `<security-constraint>` not the filter `<url-pattern>`).

26

Managing Microsoft Office Integration

This chapter provides an overview of system administrator tasks required to configure Microsoft Office integration with Oracle WebCenter Portal 11g (11.1.1.1.0) and later. For a description of how Microsoft Office can be used with Webcenter Portal, see the "Working with Microsoft Office and Explorer Integration" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

This chapter includes the following topics:

- Section 26.1, "About Microsoft Office Integration"
- Section 26.2, "Configuring Microsoft Office Integration"
- Section 26.3, "Configuring Non-SSL Integrations"
- Section 26.4, "Troubleshooting"

Permissions: To perform the tasks in this chapter, you must be granted the following roles:

- **WebLogic Server:** `Admin` role granted through the Oracle WebLogic Server Administration Console.
- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.

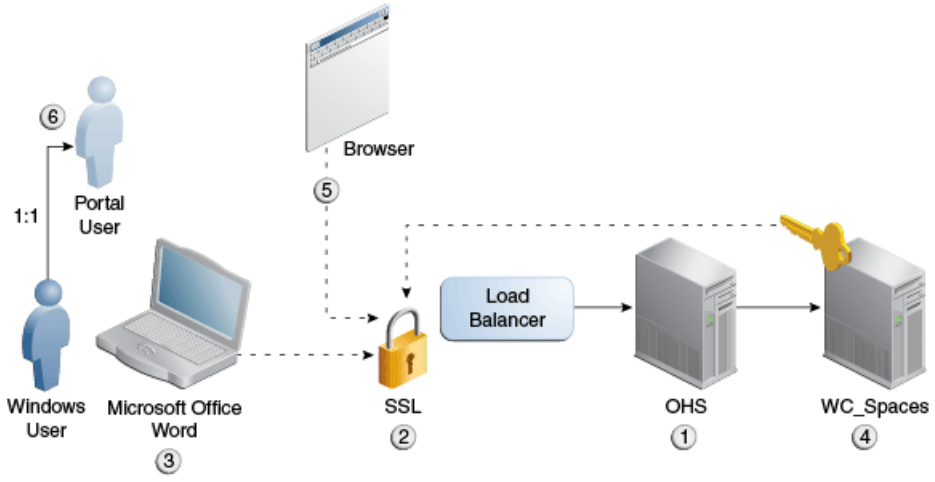
See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

26.1 About Microsoft Office Integration

Figure 26–1 shows a typical Microsoft Office integration topology with notes describing configuration concerns specific to each component in the topology. For an end-to-end description of how to configure Microsoft Office integration, see Section 26.2, "Configuring Microsoft Office Integration."

After configuring Microsoft Office integration you can interact with Microsoft Office and Microsoft Office Enterprise Edition applications from within your WebCenter Portal environment. Refer to the matrix in the "Working with Microsoft Office and Explorer Integration" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal* for the activities that are supported for each Windows version.

Figure 26–1 Microsoft Office Integration Topology



1. **Oracle HTTP Server and load balancer**

   OHS (or the load balancer) must be properly configured so that requests are routed to the Sharepoint servlet. If single sign-on is being used, you must create a virtual host that is not protected by SSO as described in Section 33.6, "Configuring SSO with Virtual Hosts." This should be done on the edge server of the topology (i.e., either the load balancer or OHS). Note that the `-Dnon_sso*` java parameters must be set to point to the non-SSO protected virtual host as described in Section 33.6.5, "Configuring WebCenter Portal for Virtual Hosts."

2. **SSL enabled entry point**

   SSL must be configured for either the load balancer or OHS, whichever is the edge server of the topology.

3. **Microsoft Office client**

   Although not using SSL imposes a security risk (in that user credentials are passed without encryption) and is strongly discouraged, you can configure each client machine's registry to allow Microsoft Office to authenticate over HTTP. For more information, see Section 26.3, "Configuring Non-SSL Integrations."

4. **WC_Spaces managed server**

   Document the applicable JVM arguments, and review specifically which ones are needed and under what conditions. For more information, refer to step 2 in Section 26.2, "Configuring Microsoft Office Integration."

   Note that if SSL is enabled on the edge server (either OHS or a load balancer), the Trusted Certificate of the SSL certificate of the edge server must be imported into the `WC_Spaces` server's keystore (see Section 26.4.3, "Using SSL - Document Cannot be Checked Out").

5. **Internet Explorer or supported browser**

   For Internet Explorer, ActiveX must be enabled. For browsers other than Internet Explorer, such as Firefox and Google Chrome, the Java plug-in must be installed. For more information, see Section 26.4.1, "Clicking Edit with Office Does Not Invoke Word."

6. **Windows/WebCenter Portal user accounts**

There must be a 1:1 relationship between Windows user accounts and WebCenter Portal login accounts. Due to the way in which integration with Microsoft Office works, WebCenter Portal user accounts must be uniquely associated with Windows user accounts. For Windows 7 in particular, the Windows 7 WebClient caches user credentials in the Windows 7 user context and consequently cannot support more than one WebCenter Portal user per Windows 7 user. For more information, see Section 26.4.2, "Problem Editing Documents from Document Library in Windows 7."

26.2 Configuring Microsoft Office Integration

This section describes how to configure Microsoft Office clients for desktop integration. Prior to following these configuration steps you should already have:

- Installed the Web Tier (Oracle HTTP Server) in front of Oracle WebCenter. For more information about installing the Oracle HTTP Server, see the "Installing and Configuring Oracle Web Tier" section in *Oracle Fusion Middleware Installation Guide for Oracle Web Tier*.
- Configured and enabled SSL on the Oracle HTTP Server (or the Load Balancing Router, if one is being used). SSL setup is mandatory if you are using Microsoft Office 2010 for desktop integration. SSL setup is recommend but not mandatory if using Microsoft Office 2007. For more information, see the "Securing the Browser Connection to WebCenter Portal with SSL" section in *Oracle Fusion Middleware Administering Oracle WebCenter Portal*.
- Imported the public certificate of the SSL certificate being used to the WebLogic Trust Store if the certificate is not one of the well known certificate authorities that is already seeded in `cacerts` or the WebLogic default Trust Store. For more information, see the "Securing the Browser Connection to Spaces with SSL" section in *Oracle Fusion Middleware Administering Oracle WebCenter Portal*.
- When WebCenter Portal is configured with OAM, the OAM administrator should have added the following resource URLs with their Protection Level set to `Excluded`:

  ```
  /wcsdocs*
  /wcsdocs/.../*
  /_vti_*
  ```

  For more information about setting resource URLs, see the "Adding and Managing Resource Definitions for Use in Policies" section in *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager with Oracle Security Token Service*.

  The OAM protection invoked within the OHS configuration must be specifically applied to the main connection port and not to the secondary unprotected SharePoint port we are configuring in the steps below (`4444` in the example). You must remove the Oblix values from `webgate.conf` (or `httpd.conf` in some cases) and replace them within the valid Virtual Host container for the main WebCenter connection (port `80` or `7777`). Then, creating the new virtual hosts in steps 1 and 2 below will create a port (`4444` in the example) that the SharePoint protocols can use to communicate without OAM SSO. For more information, see Section 33.6.4, "Configuring Virtual Hosts for OAM 11g."

Note: WebCenter Portal integration with Microsoft Office follows the model established by Microsoft for Microsoft desktop applications interacting with a SharePoint server. For WebCenter Portal integration, the `WC_Spaces` server emulates the SharePoint server's role in that model.

On the client side, the logged in Windows user may be associated with the user account used to log into WebCenter Portal, so it is important to avoid logging into multiple WebCenter Portal accounts with the same Windows user account. In particular, the Windows 7 WebClient service caches credentials used to log in to the emulated SharePoint service endpoints, so it is not possible to support various login accounts to WebCenter Portal from the same Windows user account without unintended consequences. See the troubleshooting note in Section 26.4.2, "Problem Editing Documents from Document Library in Windows 7" for more information.

To configure WebCenter Portal for desktop integration:

1. Ensure that the following mappings exist in the `webtier mod_wl_ohs.conf` file, which is located under the *`OHS_HOME`*`/Oracle_WT1/instances/`*`instance1`*`/config/OHS/ohs1` directory:

```
<Location /wcsdocs>
         SetHandler weblogic-handler
         WeblogicHost webcenter.example.com
         WeblogicPort 8888
</Location>

<Location /_vti_bin>
         SetHandler weblogic-handler
         WeblogicHost webcenter.example.com
         WeblogicPort 8888
</Location>
```

   Where, *`webcenter.example.com`* refers to the host on which WebCenter Portal is installed.

   For an example the OHS `mod_wl_ohs.conf` file, see Appendix B, "Oracle HTTP Server Configuration for WebCenter Portal."

2. If your environment is a cluster, it is recommended that you use the virtual host setup to route to the SharePoint root application.

   Note: When you have a single node setup, there is no need for a virtual host even if SSO is configured.

   In a cluster environment, ensure the following entries are present in the `httpd.conf` file, which is located under the *`OHS_HOME`*`/Oracle_WT1/instances/`*`instance1`*`/config/OHS/ohs1/` directory:

```
NameVirtualHost *:7777

<VirtualHost *:7777>
  ServerName webtier.example.com
```

```
</VirtualHost>

<VirtualHost *:7777>
  ServerName webtier-spaces.example.com
  <Location  />
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8888
  </Location>
  <Location /webcenter>
      Deny from all
  </Location>
  <Location /webcenterhelp>
      Deny from all
  </Location>
  <Location /rest>
      Deny from all
  </Location>
</VirtualHost>
```

Where:

- *webtier.example.com* refers to the OHS host.
- *webtier-spaces.example.com* refers to the virtual host. Ensure that you update the DNS with entries for *webtier-spaces.example.com*.
- *webcenter.example.com* refers to the host that has the `WC_Spaces` managed server installed.

If your environment has SSO set up, configure virtual hosts such that they can bypass SSO. For more information, see Section 33.6, "Configuring SSO with Virtual Hosts."

3. Add the following required parameters to *domain_home*`/bin/setDomainEnv.sh` (on UNIX) or *domain_home*`\bin\setDomainEnv.cmd` (on Windows):

```
EXTRA_JAVA_PROPERTIES=
"${EXTRA_JAVA_PROPERTIES} -Dnon_sso_protocol=http
-Dnon_sso_host=webcenter.example.com -Dnon_sso_port=8888
-Dsso_base_url=http://webtier.example.com:7777"
export EXTRA_JAVA_PROPERTIES
```

Where:

- `non_sso_protocol` is the protocol of the URL used to access the `WC_Spaces` managed server from Microsoft Office applications.
- `non_sso_host` is the host that points to the `WC_Spaces` managed server (that is *webcenter.example.com*) or the virtual host (that is *webtier-spaces.example.com*), if it is set up.
- `non_sso_port` is the host port that points to the `WC_Spaces` managed server port 8888, or to the virtual host port 7777, if it is set up.
- `sso_base_url` is the URL to access SSO or OHS, which is often the same as the one used by WebCenter Portal.

26.3 Configuring Non-SSL Integrations

For installations that for one reason or another do not configure SSL on the OHS or Load Balancer, you must configure client registry information for each client to override the restrictions built in to the Microsoft Office products. Refer to the following Microsoft support site and follow the instructions below. Note that the instructions may differ slightly between Windows versions:

`http://support.microsoft.com/kb/2123563`

26.4 Troubleshooting

This section includes the following sub-sections:

26.4.1 Clicking Edit with Office Does Not Invoke Word

Problem

Edit with Microsoft Office feature does not start Word or associated Office application when used with a browser other than Internet Explorer.

Solution

The Java plug-in is required for Microsoft Office integration to work with non-IE browsers. Check that you have the Java plug-in enabled in your browser. Refer to your browser's documentation for instructions for installing the Java plug-in.

26.4.2 Problem Editing Documents from Document Library in Windows 7

Problem

The first user logging into Windows 7 is able to use the Microsoft Office integration feature without any issues. However, subsequent users logging into WebCenter Portal on the same desktop in the same Windows 7 login may experience issues, especially when checking in and checking out documents. These issues may persist even though all browsers and Microsoft Office have been shut down.

The problem is due to the WebClient service, a Windows native service that allows the operating system to make HTTP and WebDAV requests, which caches the first credential. The WebClient service is intended to be used by other Windows features (for example, when a user adds a network location) and it makes sense that it caches the credential rather than asking for it every time it's accessed. However, the credential is cached in the context of the Windows 7 logged in user; it is not tied to the Spaces login.

Consequently, the WebClient service sends a request to the WebCenter Portal Document Service using the first user's credentials. For the second user, this will cause an issue because the WebCenter security model may prevent the first user credential from accessing the document (if it doesn't have the necessary rights) on the folder

being accessed by the second user. Even if it succeeds, it will appear as if the first user did the check out.

Solution

To fix the problem you can:

- Reboot the machine before a second user starts using the feature.
- Log out of Windows and log in using a different Windows 7 user.
- Restart the WebClient service:

The problem with these fixes is that they require system administrator privileges, and as the WebClient service is used by other Windows 7 features, stopping it may affect them. The easiest way to avoid this problem is to not share the same desktop and Windows 7 login across multiple WebCenter Portal or Fusion Applications user accounts.

26.4.3 Using SSL - Document Cannot be Checked Out

Problem

After clicking **Edit with Office** a dialog appears indicating that the document could not be checked out. After several login challenges, Microsoft Office opens but the document is in Read-only mode and is not checked out.

Solution

This problem relates specifically to the following environment:

- The browser is Internet Explorer
- OHS or a load balancer is set up in front of the `WC_Spaces` server
- SSL is enabled on OHS and terminates at OHS (i.e., the connection from OHS to the `WC_Spaces` server is non-SSL)

This symptom occurs because there is a second HTTPS request from the backend (`WC_Spaces` server) to the OHS (or load balancer), which throws a SSL Key exception because the `WC_Spaces` server is not trusted. This is the request that is responsible for doing the document check-out.

To resolve this issue:

1. Import the Trusted Certificate from the OHS or load balancer to the `WC_Spaces` server.
2. Export the Trusted Certificate from the OHS Wallet following the steps below:
   a. Log into the Fusion Middleware Control instance that manages OHS.
   b. Select **Web Tier > ohs1**.
   c. From the OHS drop-down list, select **Security > Wallets**.
   d. Click **default**.
   e. Select `CN="\"Self-Signed Certificate for ohs1 \"", OU=OAS, O=ORACLE, L=REDWOODSHORES, ST=CA, C=US`
   f. Click **Export**.
   g. Save the file (for example, as "ohsTrustedCertificate").
   h. Copy the file to the local disk of the `WC_Spaces` server.

3. Import the OHS Trusted Certificate to the WLS `DemoTrust.jks` using the following keytool command:

```
JAVA_HOME/bin/keytool -importcert -v -alias ohscert -file
/mycert/ohsTrustedCertificate -keystore
/my_mw_home/wlserver_10.3/server/lib/DemoTrust.jks -storepass
DemoTrustKeyStorePassPhrase
```

   where `DemoTrustKeyStorePassPhrase` is the default password for the `DemoTrust.jks`.

   The path for the keystore can be found by:

   a. Logging into the WLS Console.

   b. Selecting **Environment > Servers > WC_Spaces**.

   c. Opening the Configuration tab and then selecting **Keystores**.

26.4.4 Microsoft Office Task Pane Only Shows a Single Tab

Problem

After clicking **Edit with Office** on a document in Internet Explorer Microsoft Office launches with only one tab on the Task Pane.

Solution

This problem occurs because the **Use my local drafts folder** option was selected in Internet Explorer when the document was opened, resulting in the file being copied to the user's local folder rather than connected to the server. If the file is not subsequently checked in, the same symptom will occur for other users trying to edit the document with other browsers such as Firefox or Chrome. To avoid this problem, be sure all users uncheck the **Use my local drafts folder** option in Internet Explorer when prompted. For all other browsers, be sure that users use **Options > Save** with each of the MS Office applications.

26.4.5 Unable to Connect to Microsoft Office Using Firefox

Problem

Unable to connect to Microsoft Office applications from WebCenter Portal when using Firefox.

Solution

Due to security issues with Java 7, Firefox is now blocking the Java Platform Plug-In **even when it appears to be enabled in the plug-ins list**, which will effectively disable Microsoft Office integration.

In order to use Java and Microsoft Office integration in Firefox, you must now additionally click the plug-in icon (see Figure 26–2):

Figure 26–2 Plug-in Icon

at the top left of the browser adjacent to the URL bar, and explicitly enable Java for the site you want.

Part VI

Monitoring

This part of *Oracle Fusion Middleware Administering Oracle WebCenter Portal* presents information about monitoring WebCenter Portal and Portal Framework applications using Oracle Enterprise Manager Fusion Middleware Console.

Part VI contains the following chapters:

27

Monitoring Oracle WebCenter Portal Performance

This chapter describes the range of performance metrics available for WebCenter Portal and Portal Framework applications and how to monitor them through Fusion Middleware Control. It also describes how to troubleshoot issues by analyzing information that is recorded in diagnostic log files. Administrators who monitor WebCenter Portal or Portal Framework applications regularly will learn to recognize trends as they develop and prevent performance problems in the future.

This chapter includes the following topics:

- Section 27.1, "Understanding Oracle WebCenter Portal Performance Metrics"
- Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control"
- Section 27.3, "Customizing Key Performance Metric Thresholds and Collection"
- Section 27.4, "Diagnosing and Resolving Performance Issues with Oracle WebCenter Portal"
- Section 27.5, "Tuning Oracle WebCenter Portal Performance"

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin`, `Operator`, or `Monitor` role through the Oracle WebLogic Server Administration Console.

See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

27.1 Understanding Oracle WebCenter Portal Performance Metrics

Through Fusion Middleware Control, administrators can monitor the performance and availability of all the components, tools, and services that make up WebCenter Portal and Portal Framework applications, as well as the application as a whole. To access Oracle WebCenter Portal metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

To make best use of the information displayed it is important that you understand how performance metrics are calculated and what they mean. All Oracle WebCenter Portal's performance metrics are listed and described here for your reference. Some applications (such as WebCenter Portal) might use the full range of social networking, personal productivity, and collaboration metrics listed, while others may only use one or more of these features.

This section includes the following topics:

27.1.1 Understanding Oracle WebCenter Portal Metric Collection

Performance metrics are automatically enabled for Oracle WebCenter Portal and display in Fusion Middleware Control. You do not need to set options or perform any extra configuration to collect performance metrics for WebCenter Portal or a Portal Framework application. If you encounter a problem, such as, an application running slowly or hanging, you can find out more about the problem by investigating performance metrics, in real-time, through Fusion Middleware Control.

This section describes the different ways Oracle WebCenter Portal collects and presents metric data:

- Metric Collection: Since Startup
- Metric Collection: Recent History
- Metric Collection: Last 'N' Samples

27.1.1.1 Metric Collection: Since Startup

At any given time, real-time metrics are available for the duration for which the WebLogic Server hosting WebCenter Portal or your Portal Framework application is up and running. Real-time metrics that are collected or aggregated since the startup of the container are displayed on Oracle WebCenter Portal metric pages under the heading **Since Startup**. These metrics provide data aggregated over the lifetime of the WebLogic Server. The aggregated data enables you to understand overall system performance and compare the performance of recent requests shown in **Recent History**.

For example, consider WebCenter Portal deployed on a managed server that was started 4 hours ago. During that time, WebCenter Portal serviced 10,000 portlet requests with a total response time of 500, 000 ms. For this scenario, **Since Startup** metrics for portlets show:

- **Since Startup: Invocations** (count) - 10000
- **Since Startup: Average Time** (ms) - 50

Note: Metric collection starts afresh after the container is restarted. Data collected before the restart becomes unavailable.

27.1.1.2 Metric Collection: Recent History

In addition to **Since Startup** metrics, Oracle WebCenter Portal reports metrics for requests serviced in the last 10 to 15 minutes as **Recent History** metrics. To do this, Oracle WebCenter Portal takes regular snapshots of real time metrics at an internal frequency. These metric snapshots are used to calculate the "delta" time spent performing service requests in the last 10 to 15 minutes and this data displays as **Recent History** metrics. Since Recent History metrics only aggregate data for the last 10-15 minutes, this information is useful if you want to investigate ongoing performance/availability issues.

If you compare Recent Metrics to Since Startup metrics you can gauge how the system characteristics have changed, compared to overall system availability/performance.

For example, consider a system that has been up and running for 2 days. During that time, Oracle WebCenter Portal recorded that the total time spent servicing 100, 000 portlet requests was 5 000 000 ms. The system starts to experience performance issues, that is, in the last 10-15 minutes, 100 portlet requests took a total time of 3 000 000 ms. In this scenario, the *average response time* reported "Since Startup" is quite low and would not indicate a performance issue (5 000 000ms/100 000 = 50ms). However, the same Recent History metric is considerably higher (3000000ms/100 = 30 seconds) which immediately tells the administrator that performance degraded recently. A quick comparison of "Recent History" with the corresponding "Since Startup" metric can clearly show whether or not the recent metric data is normal and in this case shows there is currently a problem with the system.

Recent History metrics can also help you prioritize which areas to investigate and which areas you can ignore when performance issues arise. For example, if an ongoing performance issue is reported and Recent History metrics for a particular component shows a value of 0, it indicates that the component has not been used in the last 10-15 minutes. Similarly, if the "Average Response Time" value is small and the "Invocation" count is low, the component may not be contributing to the performance problem. In such cases, administrators can investigate other areas.

Typically, Recent History shows data for the most recent 10-15 minutes. However, there are situations when the data does not reflect the last 10-15 minutes:

- If the WebLogic Server has just started up, and has been running for less than 10-15 minutes, then Recent History shows data for the duration for which the server has been up and running.
- If one or more tools or services are not accessed for an extended period of time, then older metric snapshots slowly age out. In such cases, metric data is no longer available for the last 10-15 minutes so Recent History metrics cannot calculate the delta time spent in performing service requests that occurred in last 10-15 minutes. When this happens, the Recent History data can show the same values as the Since Startup metrics. When the tool or service is used again, metric snapshots for it resume. After enough recent data is available, the Recent History metrics again start to display metrics for the last 10-15 minutes.

  Most live environments are not idle for extended periods, so recent metric collection is rarely suspended due to inactivity. However, if you have a test environment that is used intermittently or not used for a while, you might notice recent metric collection stop temporarily, as described here.

27.1.1.3 Metric Collection: Last 'N' Samples

Since Startup and **Recent History** metrics calculate performance over a specific duration, and show aggregated metrics for that duration. In addition to these, Oracle WebCenter Portal collects and reports per-request performance information for a range of *key WebCenter Portal metrics*. Such metrics allow you to look at the success and response time of each request individually, without considering previous requests. Out-of-the-box, the last *100* samples are used to calculate key metric performance/availability but you can increase or decrease the sample set to suit your installation.

For example, if 10 out of the last 100 page requests failed, page availability is calculated as 90%. If you reduce the sample set to 50 and 10 pages fail, page availability is reported to be 80%.

The examples show how the sample set size can effect the performance reports. The value you select is up to you but if you increase the number of samples, consider the additional memory requirements since the last 'N' metric samples are maintained in memory. Oracle recommends a few hundred samples at most.

To change the number of samples used to report key performance metrics in your installation, see Section 27.3.6, "Configuring the Number of Samples Used to Calculate Key Performance Metrics."

To find out more about Oracle WebCenter Portal's key performance metrics and thresholds, refer to Section 27.1.2, "Understanding the Key Performance Metrics."

27.1.2 Understanding the Key Performance Metrics

Diagnosing the availability and performance of WebCenter Portal or Portal Framework applications typically requires that you look at various important metrics across multiple components such as the JVM, the WebLogic Server, as well as the application.

To help you quickly identity and diagnose issues that can impact WebCenter Portal or Portal Framework application performance, Oracle WebCenter Portal collects the last 'N' samples for a range of "*key performance metrics*" and exposes them in Fusion Middleware Control. To access key performance metric information for your application, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

Thresholds determine when a performance alert or warning is triggered. Allowing you to set threshold values that represent suitable boundaries for your Oracle WebCenter Portal system, ensures that you obtain relevant performance alerts in Enterprise Manager Fusion Middleware Control. When key performance metrics are "out of bounds" with respect to their configured thresholds they are easy to find in Fusion Middleware Control as they appear color-coded. For more information about thresholds, see Section 27.3, "Customizing Key Performance Metric Thresholds and Collection."

You do not need to specifically set thresholds for metrics, such as "availabilty", that report success or failure. For example, document download failures automatically appear color coded "red" and successful downloads always appear "green".

Oracle WebCenter Portal allows you to manage warning thresholds for the key performance metrics described in Table 27–1:

Table 27–1 Key Performance Metric Collection

Component	Key Performance Metric	Metric Sampling
WebCenter Portal or Portal Framework Application	Active Sessions	1 sample every X minutes
WebCenter Portal or Portal Framework Application - **Pages**	Page Response Time	Per Request
WebCenter Portal or Portal Framework Application - **Documents**	Download Throughput Acceptable Download Time	Per Request
WebCenter Portal or Portal Framework Application - **Documents**	Upload Throughput Acceptable Upload Time	Per Request
WebCenter Portal or Portal Framework Application - **Portlets**	Portlet Response Time	Per Request
JVM	CPU Usage	1 sample every X minutes
JVM	Heap Usage	1 sample every X minutes
JVM	Garbage Collection Rate	1 sample every X minutes
JVM	Average Garbage Collection Time	1 sample every X minutes
WebLogic Server	Active Execute Threads	1 sample every X minutes
WebLogic Server	Execute Threads Idle Count	1 sample every X minutes
WebLogic Server	Hogging Execute Threads	1 sample every X minutes
WebLogic Server	Open JDBC Sessions	1 sample every X minutes

Oracle WebCenter Portal captures end-user requests for pages, portlets, and documents and a metric sample is collected for each request. For example, if user A accesses page X, both the *availability* of page X (success/fail metric) and the *response time* of the request is captured by Oracle WebCenter Portal. Metric samples that take longer than a configured metric alert threshold or fail, show "red" in Fusion Middleware Control to immediately alert administrators when issues arise.

Other metrics, such as JVM and WebLogic Server metrics, are collected at a pre-defined frequency. Out-of-the-box, the sample frequency is 1 sample every 5 minutes but you can customize this value if required. For details, see Section 27.3.5, "Configuring the Frequency of WebLogic Server Health Checks."

The total number of samples that Oracle WebCenter Portal collects is configurable too, as described in Section 27.3.6, "Configuring the Number of Samples Used to Calculate Key Performance Metrics." The default sample set is 100 samples. Since there is a memory cost to maintain metric samples, do not specify an excessive number of samples; Oracle recommends a few hundred at most.

Oracle WebCenter Portal's key performance metrics are specifically selected to help administrators quickly identity and diagnose common issues that can impact WebCenter Portal or Portal Framework application performance. You can view all key performance metric data from your application's home page in Fusion Middleware Control.

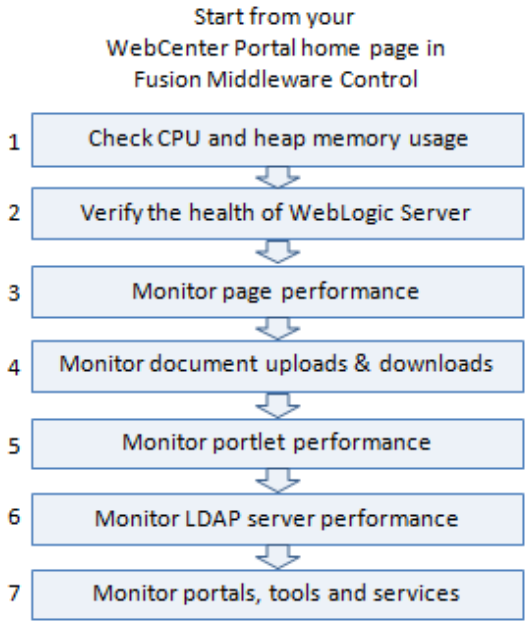
27.1.3 Using Key Performance Metric Data to Analyze and Diagnose System Health

If you monitor WebCenter Portal or your Portal Framework application regularly, you will learn to recognize trends as they develop and prevent performance problems in the future. The best place to start is your application's home page in Enterprise Manager Fusion Middleware Control. The home page displays status, performance, availability, and other key metrics for the various components, tools, and services that make up your application, as well as the WebLogic Server on which the application is deployed.

If you are new to Oracle WebCenter Portal, use the information in this section to better understand how to use the information displayed through Fusion Middleware Control to identify and diagnose issues.

Figure 27–1 presents high-level steps for monitoring the out-of-the-box application *WebCenter Portal*.

Figure 27–1 Analyzing System Health for WebCenter Portal - Main Steps



Note:

- Steps 4 and 5 only apply if your application utilizes portlets or document features.
- Bar charts appear grey if a feature is not used.
- Line charts require at least 3 data points before they start to show data.

Figure 27–2 Analyzing System Health from the WebCenter Portal Home Page

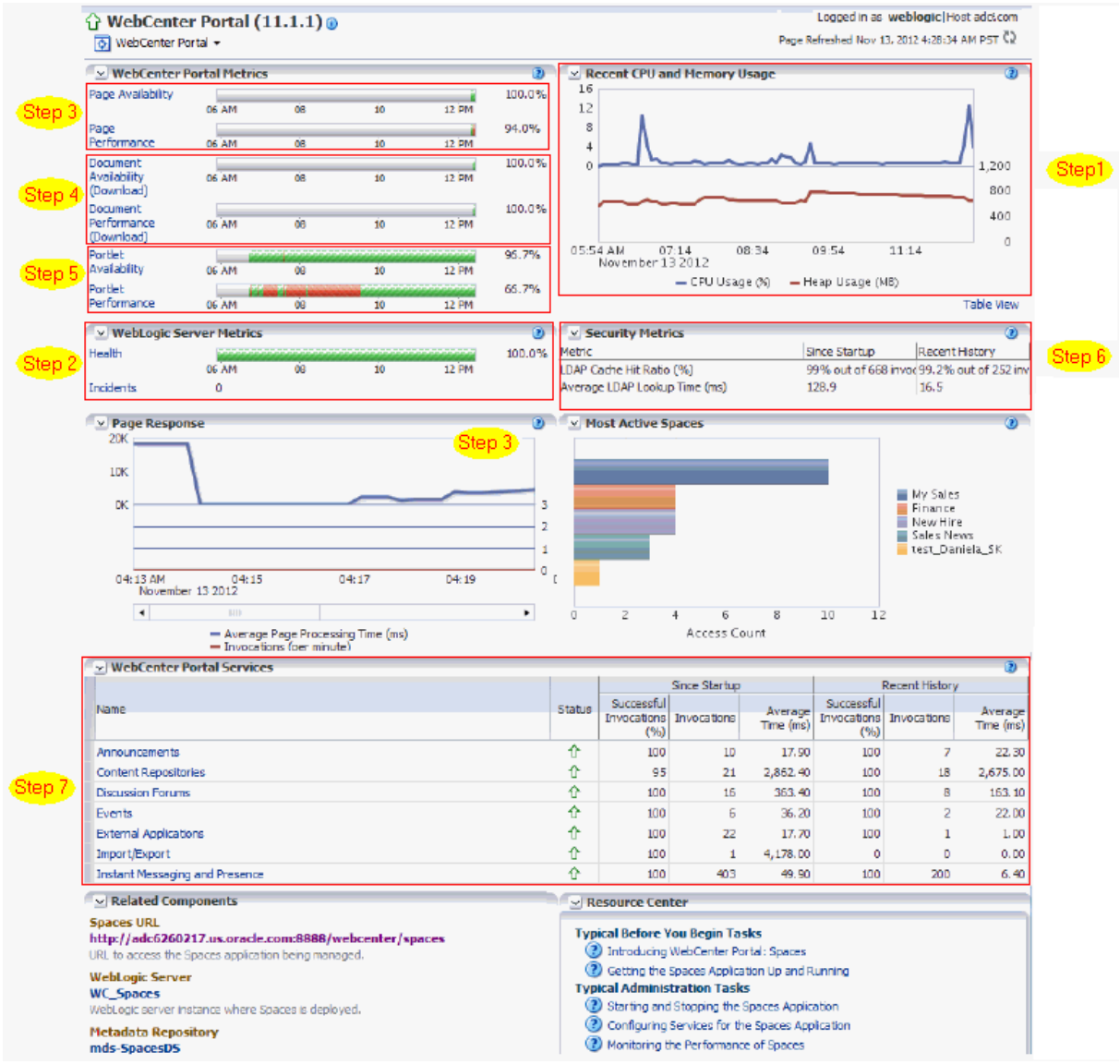


Table 27–2 Analyzing System Health - Step by Step

Step	Description
Navigate to the home page for WebCenter Portal or Portal Framework application	Use Enterprise Manager Fusion Middleware Control to monitor the performance of your portal application. The best place to start is your application's home page: ▪ Section 6.2, "Navigating to the Home Page for WebCenter Portal" ▪ Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
1 Check CPU and heap memory usage	Overall performance deteriorates when CPU or memory usage is too high so its important that you always look at the CPU and memory metrics *before* looking at any other Oracle WebCenter Portal-specific metric. Check the **Recent CPU and Memory Usage** charts to see the current usage trend: ▪ **High CPU usage**? Occasional spikes in CPU usage is normal but if CPU usage remains high (85-90%) over a long period of time, it normally indicates there is an issue with CPU. To troubleshoot CPU issues, see: Section 27.1.8, "Understanding WebLogic Server Metrics" Section G.4.2.3, "Monitoring Java Virtual Machine (JVM) Usage" ▪ **High memory usage**? When the chart shows that memory is close to the maximum heap size and the trend is not downwards, take some memory dumps to further analyze the cause. To access maximum heap size information: 1. Log in to WebLogic Server Administration Console. 2. Navigate to: **Environment**> **Servers**> <*managed_server name*> 3. Click **Monitoring**> **Performance** tab. 4. Look at "**Heap Size Max**". See Section G.4.5.3, "Troubleshooting Slow Requests Using JFR Recordings." See Section G.4.2.1, "Verifying System Resources (CPU and Memory)." **Note:** For Portal Framework applications, select **Application Deployment >WebCenter Portal >Recent WebLogic Server Metrics** to display the **Recent CPU and Memory Usage** chart. **Next Step:** If the charts indicate that CPU and memory usages are normal, verify the health of the WebLogic Server.

Table 27–2 (Cont.) Analyzing System Health - Step by Step

Step	Description
2 Verify the health of WebLogic Server	Look in the **WebLogic Server Metrics** region: ■ **Health** - The bar chart summarizes recent WebLogic Server health, as reported by the Oracle WebLogic Server self-health monitoring feature. For example, if 10 out of the last 100 WebLogic Server health checks fail (do not report OK), WebLogic Server health is shown as 90%. Click the **Health** link to navigate to more detail on the Recent WebLogic Server Metrics page. ■ **Incidents** - The number of times WebLogic Server metrics, such as CPU usage, memory usage, thread count, number of JDBC connections, session metrics, and so on, exceed threshold settings. Click the **Incidents** link to diagnose incidents further. The actions you take next depend on the metric data. For example, if there are hogging threads, you can take thread dumps. If JDBC connections are exceeding limits, you can analyze further for connection leaks. If the garbage collection rate is exceeding limits, you can take heap dumps, an so on. For details, see Section 27.1.8, "Understanding WebLogic Server Metrics" and Section G.4, "Troubleshooting Oracle WebCenter Portal Performance Issues." Out-of-bound metrics show "red" in charts and "orange" in the Health Metrics table. Examine all occurrences of such situations by scanning the diagnostic logs. In-memory information is limited to "N" metric samples, but the logs store much more historical information about how often a problem is happening, as well as additional contextual information, such as which user. Here is sample message: `[WC_Spaces] [WARNING] [WCS-69252] [oracle.webcenter.system-management] [tid: oracle.webcenter.DefaultTimer] [ecid: 0000JhEX92mEgKG_Ix8Dyf1Ghz32000002,0] [APP: webcenter#11.1.1.4.0] wlsCpuUsage: 21.92100394175851 % of WebLogicServer is out-of-bounds` **Tip:** You can use Fusion Middleware Control to locate all messages of this type by searching the message type, message code, and other string pattern details. See Section 28.2, "Viewing and Configuring Log Information." By default, a warning thresholds is only set for **CPU Usage** but you can configure thresholds for other key WebLogic Server metrics, such as **Heap Memory Usage**. See Section 27.3.3, "Configuring Thresholds for Key Metrics." Look at diagnostics logs for errors, failures, and any configuration or network issues. If an issue relates to another backend server, such as, WebCenter Content and SOA, verify the JVM/WebLogic Server health (CPU, heap, threads, and so on) for those managed servers too. Similarly, investigate WebLogic Server health for other managed servers in your WebCenter Portal installation such as WC_Portlet, WC_Utilities, and WC_Collaboration. **Note:** For Portal Framework applications, select **Application Deployment >WebCenter Portal >Recent WebLogic Server Metrics** to display health metrics. **Next Step**: If the charts indicate that WebLogic Server is performing within thresholds, verify the health of your WebCenter Portal application.

Table 27–2 (Cont.) Analyzing System Health - Step by Step

Step	Description
3 Monitor page performance	Look at the **WebCenter Portal Metrics** section at the top of the home page. Review the page availability/performance charts to see whether page requests are currently responding as expected. Drill down to more detail to investigate issues relating to recent page requests. Use the Sort Ascending/Descending arrows for the **Time** and **Page Name** columns to see whether a pattern is emerging for a specific page or set of pages, or whether performance spikes appear to be more random. Out-of-bound metrics show "red" in charts and "orange" in the Page Metrics table. For details, see Section 27.1.5, "Understanding Page Request Metrics." Examine all occurrences of such situations by scanning the diagnostic logs. In-memory information is limited to "N" metric samples, but the logs store much more historical information about how often a problem is happening, as well as additional contextual information, such as which user. Here is sample message: `[WC_Spaces] [WARNING] [WCS-69251] [oracle.webcenter.system-management] [tid: [ACTIVE].ExecuteThread: '4' for queue: 'weblogic.kernel.Default (self-tuning)'] [userId: weblogic] [ecid: 6356ef0164cbad47:3fe105c5:13b4e847973:-8000-0000000000000031,0] [APP: webcenter#11.1.1.4.0] [DSID: 0000JhEYRT^EgKG_Ix8Dyf1Ghz32000005] pageResponseTime: 22223 ms of PersonalSpace/Activities is out-of-bounds` **Tip:** You can use Fusion Middleware Control to locate all messages of this type by searching the message type, message code, and other string pattern details. See Section 28.2, "Viewing and Configuring Log Information." Identify individual pages that are not performing. For details, see Section G.4.3, "How to Identify Slow Pages." Navigate to the "**Overall Page Metrics**" page to see how this page has performed historically (since startup, and last 10-15 minutes). Has it always been slow? For pages that are failing, see Section G.4.5, "How to Troubleshoot Slow Page Requests." **Note:** For Portal Framework applications, select **Application Deployment >WebCenter Portal >Recent Page Metrics** to display page metrics. **Next Step:** If the charts indicate that page requests are performing within thresholds, verify document upload/download performance.

Table 27–2 (Cont.) Analyzing System Health - Step by Step

Step	Description
4. Monitor document uploads and downloads	Look at the **WebCenter Portal Metrics** section at the top of the home page. Review the document availability/performance charts to see whether document downloads are currently performing as expected. Drill down to more detail to investigate issues relating to recent document requests. Out-of-bound metrics show "red" in charts and "orange" in the Document Metrics table. For details, see Section 27.1.6, "Understanding Document Metrics." Examine all occurrences of such situations by scanning the diagnostic logs. In-memory information is limited to "N" metric samples, but the logs store much more historical information about how often a problem is happening, as well as additional contextual information, such as which user. Here is sample message: `[WC_Spaces] [WARNING] [WCS-69255] [oracle.webcenter.system-management] [tid: [ACTIVE].ExecuteThread: '0' for queue: 'weblogic.kernel.Default (self-tuning)'] [userId: weblogic] [ecid: 6356ef0164cbad47:3fe105c5:13b4e847973:-8000-0000000000000060,0] [APP: webcenter#11.1.1.4.0] [DSID: 0000JhEYRT^EgKG_Ix8Dyf1Ghz32000005] downloadThroughput: 11.63793103448276 KB/sec of 3209 is out-of-bounds`  `[WC_Spaces] [WARNING] [WCS-69254] [oracle.webcenter.system-management] [tid: [ACTIVE].ExecuteThread: '0' for queue: 'weblogic.kernel.Default (self-tuning)'] [userId: weblogic] [ecid: 6356ef0164cbad47:3fe105c5:13b4e847973:-8000-0000000000000249,0] [APP: wcportal] [DSID: 0000JhEbmszEgKG_Ix8Dyf1Ghz32000009] uploadThroughput: 95.90502106741573 KB/sec of OWCSVR01USORAC011587 is out-of-bounds` **Tip:** You can use Fusion Middleware Control to locate all messages of this type by searching the message type, message code, and other string pattern details. See Section 28.2, "Viewing and Configuring Log Information." Navigate to the "**Overall Service Metrics**" page, and then select **Content Repositories** to see how documents have performed historically (since startup, and last 10-15 minutes). Have document performance deteriorated recently or always been slow? If document performance is normally within thresholds: 1. Verify JVM/WebLogic Server health for the server that is hosting WebCenter Content (CPU, heap, threads, and so on). 2. Monitor metrics for Content Server, as well as the database on which the documents are stored. 3. Directly access Content Server and issue upload/download operations to assess availability/performance. 4. Review the content repository connection to Oracle WebCenter Content Server and compare with the configuration of the Content Server. 5. Check for network connectivity issues between WebCenter Portal and Content Server. 6. Simulate document operations in WebCenter Portal, that is, perform document downloads/uploads to verify whether the problem is pervasive or intermittent. **Note:** For Portal Framework applications, select **Application Deployment >WebCenter Portal >Recent Document Metrics** to display document metrics. **Next Step:** If the charts indicate that document requests are performing within thresholds, verify portlet performance.

Table 27–2 (Cont.) Analyzing System Health - Step by Step

Step	Description
5. Monitor portlet performance	Look at the **WebCenter Portal Metrics** section at the top of the home page. Review the portlet availability/performance charts to see whether portlets are currently performing as expected. Drill down to more detail to investigate issues relating to recent portlet requests. Out-of-bound metrics show "red" in charts and "orange" in the Portlet Metrics table. For details, see Section 27.1.7, "Understanding Portlet Producer Metrics." Out-of-bound conditions are also logged in managed server diagnostic logs so you can examine all historical events, that is, more that the most recent sample set that is held in memory. For example: `[WC_Spaces] [WARNING] [WCS-69253] [oracle.webcenter.system-management] [tid: pool-3-daemon-thread-1] [userId: weblogic] [ecid: 6356ef0164cbad47:3fe105c5:13b4e847973:-8000-0000000000000088,0 :16] [APP: webcenter#11.1.1.4.0] portletResponseTime: 20523 ms of Portlet: slowRenderingPortlet from Web Producer Myjpdk is out-of-bounds.` Identify individual portlets or portlet producers that are not performing as expected. Navigate to the "**Overall Service Metrics**" page, and then select **Portlet Producers** or **Portlets** to see how these portlets/portlet producers have performed historically (since startup, and last 10-15 minutes). Has performance deteriorated recently or always been slow? If portlet performance is normally within thresholds: 1. Verify JVM/WebLogic Server health for the managed server that is hosting the portlets (for example, WC_Portlets), that is, investigate CPU, heap, threads, and so on. 2. Enter the portlet producer's URL in your browser to determine whether the producer is available. 3. Review the portlet producer's connection configuration. 4. Check for network connectivity issues between the WebCenter Portal application and the portlet producer. 5. Simulate portlet operations in WebCenter Portal, that is, view, personalize, or interact with the portlet to verify whether the problem is pervasive or intermittent. For portlets that are failing, see the "Troubleshooting Portlets" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*. **Note:** For Portal Framework applications, select **Application Deployment >WebCenter Portal >Recent Portlet Metrics** to display portlet metrics. **Next Step:** If the charts indicate that portlet requests are performing within thresholds, verify the performance of your LDAP server.
6. Monitor LDAP server performance	Look at the LDAP metrics in the **Security** section on the home page. When the server first starts up the cache hit ratio is zero and typically increases above 90% as the system warms up. For more information, see Section 27.1.9, "Understanding Security Metrics" and Section 16.9, "Configuring Cache Options for the Profile Service." Typically, the average LDAP lookup time is only a few milliseconds. If lookups are taking a long time there maybe a problem with the LDAP server or network relate issue. ▪ If you want to measure the response time from the LDAP server for a simple bind operation, run the command: `ldapbind -D "UserDN" -h ldaphost.example.com -p <port> -w <password>` To investigate further network issues, see Section G.4.2.6, "Diagnosing Network Related Problems Using tcpdump" or Section G.4.2.7, "Measuring Network Latency Using ping." If you are using Oracle Internet Directory, refer to "Oracle Internet Directory Performance Tuning" in *Oracle Fusion Middleware Performance and Tuning Guide* for advice on how to improve performance and avoid bottlenecks. For other LDAP servers, refer to the appropriate product documentation. **Next Step:** If your LDAP server is performing within thresholds, investigate other areas.

Table 27–2 (Cont.) Analyzing System Health - Step by Step

Step	Description
7. Monitor individual tools and services	Look at the **WebCenter Portal Services** section at the bottom of the home page. For details, see Section 27.1.12, "Understanding Tool and Service Metrics." Quickly see if a particular tool or service is "Down" or "Unknown". Refer to Section 27.1.12.3, "Troubleshooting Common Issues with Tools and Services" for guidance on possible causes and actions. Sort the table by **Average Time** or **Invocations** to prioritize which tool or service to focus on. Click a name to navigate to the "**Overall Service Metrics**" page. Compare **Since Startup** and **Recent History** metrics to see if performance deteriorated recently or always been slow. **Note:** For Portal Framework applications, select **Application Deployment >WebCenter Portal >Overall Service Metrics** to display metrics.

27.1.4 Understanding Some Common Performance Issues and Actions

If an Oracle WebCenter Portal metric is out-of-bounds, do the following:

- Check system resources, such as memory, CPU, network, external processes, or other factors. See Appendix G, "Troubleshooting Oracle WebCenter Portal."
- Check other metrics to see if the problem is system-wide or only in a particular tool or service.
- If the issue is related to a particular tool or component, then check if the back-end server is down or overloaded.
- If the WebLogic Server has been running for a long time, compare the **Since Startup** metrics with the **Recent History** metrics to determine if performance has recently deteriorated, and if so, by how much.
- When the status of a tool or service is *Down* or some operations do not work, then validate, test, and ping the back-end server through direct URLs. For details, refer to the "Testing Connection" section in the relevant chapter. For a list of chapters, see Part V, "Managing Tools, Portlet Producers, and External Applications".

  When you reconfigure connections to tools and services you must always restart the managed server on which WebCenter Portal or your Portal Framework application is deployed to pick up the changes. If key connection attributes change, such as a server's host/port details, connectivity to the server may be lost and the service may become unavailable until you reconfigure the connection and restart the managed server.

> **Note:** You can customize the threshold at which some key performance metrics trigger out-of-bound conditions. See Section 27.3, "Customizing Key Performance Metric Thresholds and Collection."

27.1.5 Understanding Page Request Metrics

You can monitor the availability and performance of page requests for WebCenter Portal or a Portal Framework application through Fusion Middleware Control. You can monitor recent page data and historical (overall) page data.

This section includes the following information:

- Section 27.1.5.1, "Understanding Full Page and Partial Page Metrics"
- Section 27.1.5.2, "Recent Page Metrics"
- Section 27.1.5.3, "Overall Page Metrics"

Note: The *page request* metrics discussed in this section are different from the *page operation* metrics discussed in Section 27.1.12.2.12, "Page Operation Metrics." Page operation metrics monitor page related operations such as creating pages. Whereas the page request metrics described here monitor individual page view/display requests (do not include page edit operations).

27.1.5.1 Understanding Full Page and Partial Page Metrics

Performance data is collected for full page and partial page requests. Full page metrics do not include partial page metrics.

Partial page requests display only portions of the page. Therefore, you can monitor the performance of pages within a page. Partial page refresh behavior is called partial page rendering (PPR). PPR allows only certain components on a page to be rerendered without the need to refresh the entire page. A common scenario is when an output component displays what a user has chosen or entered in an input component. Similarly, a command link or button can cause another component on the page to be rerendered without refreshing the entire page.

Partial page rendering of individual components on a page only increases partial page metrics and does not cause any change in full page metrics. For example, a calendar refresh on a page increases partial page invocations by 1, but full page invocations remain unchanged.

For more information about PPR, see the "Rerendering Partial Page Content" chapter in *Oracle Fusion Middleware Web User Interface Developer's Guide for Oracle Application Development Framework*.

27.1.5.2 Recent Page Metrics

Recent page availability and performance metrics are summarized on the home page for WebCenter Portal or your Portal Framework application (Figure 27–3 and Table 27–3). The page availability/performance charts show at a glance if page requests are slower than expected or failing.

Note: To access the home page, see Section 6.2, "Navigating to the Home Page for WebCenter Portal" or Section 6.3, "Navigating to the Home Page for Portal Framework Applications."

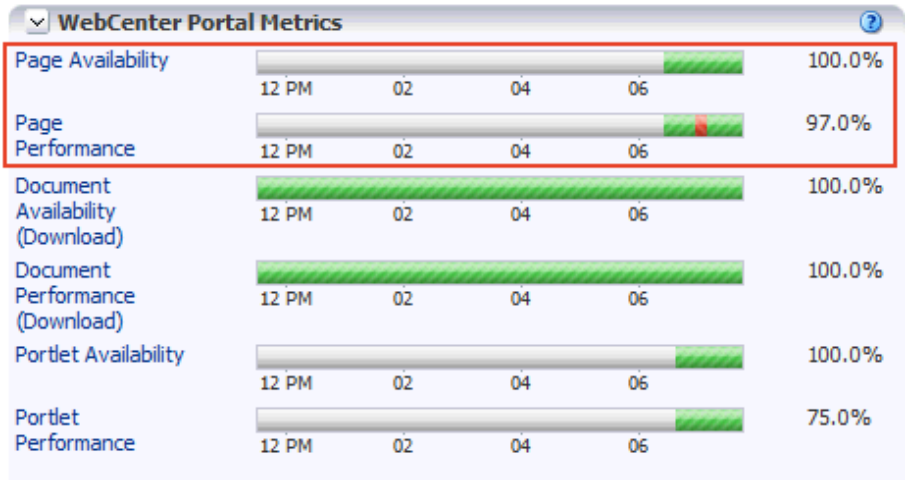
The **Page Availability** and **Page Performance** charts report availability and performance over the last 'N' page requests (by default, 'N' is 100). The time range starts with the earliest page/document/portlet request time and ends with the current time. See Section 27.3.6, "Configuring the Number of Samples Used to Calculate Key Performance Metrics."

The % value on the right shows the percentage of page requests that responded within a specific time limit. The percentage is calculated using information from the last 'N' page requests. For example, if 'N' is 100, and if 3 of the last 100 page requests exceeded the page response threshold, page performance is shown as 97%.

The bar chart status (green/red) does not change over time until the status changes, so the % performance value and the visual green/red ratio do not always match up. For example, consider a scenario where the first 5 page requests are "out of bounds", the system is idle (no page requests) for 9 hours, and then there are 95 "good" page requests within an hour. In this instance the chart displays 90% red (9 hours) and 10%

green (1 hour) but the % performance value shows 95% ('N' is 100 and 95 samples out of 100 are "good"). The mismatch occurs because the bar charts plot uniformly over time, whereas page requests are not usually uniformly distributed over time.

Figure 27–3 Recent Page Summary on the WebCenter Portal Home Page

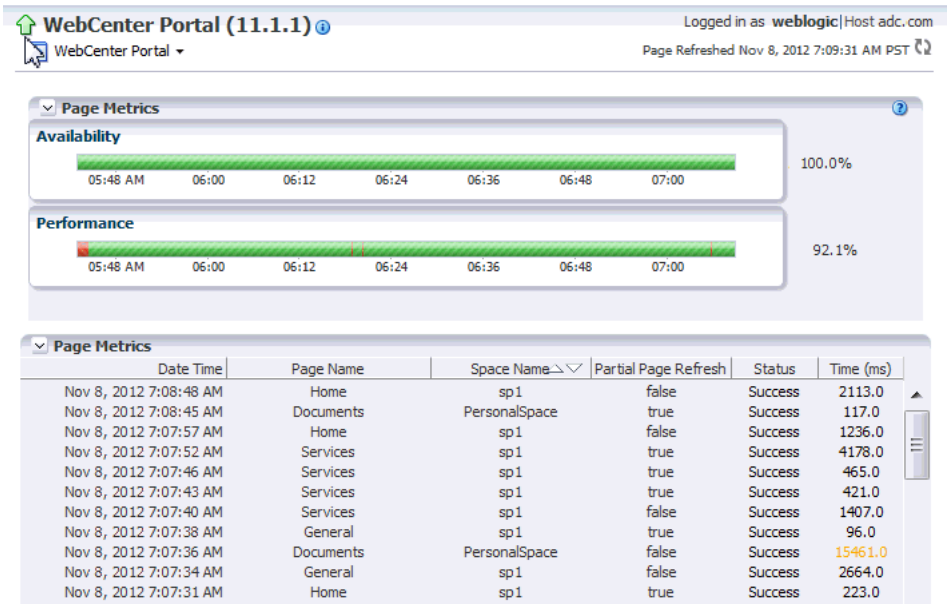


If the chart indicates issues or incidents, click the **Page Availability** or **Page Performance** link to navigate to more detailed information to diagnose the issue further (see Figure 27–4 and Table 27–3).

Use the information on the Recent Page Metrics page (Figure 27–4) to troubleshoot recent page performance issues. The page availability/performance charts at the top of the page show "red" if page requests are slower than expected or failing.

Note: Out-of-the-box, the page response threshold is 10, 000ms so pages taking longer than 10, 000ms to respond show "red" in the chart. If this threshold is not suitable for your installation you can change the threshold value. See Section 27.3, "Customizing Key Performance Metric Thresholds and Collection."

Figure 27–4 Recent Page Metrics



The charts report availability/performance over the last 'N' page requests. The time range starts with the earliest page request time and ends with the time of the last page request.

Use the information in the table to identify slow pages, that is, the name of the page and the portal to which the page belongs.

To diagnose page response issues, refer to the advice in "*Step 3. Monitor page performance*" in Table 27–2, " Analyzing System Health - Step by Step".

Table 27–3 Recent Page Request Metrics

Metric	Description
Availability	Indicates page availability over the last 'N' page requests: ▪ **Green** - Indicates successful page requests. ▪ **Red** - Indicates that a failure occurred during a page request. Look at the **Status** column in the table below to identify any page requests that fail. ▪ % - Percentage of page requests that succeeded. The percentage is calculated using status information from the last 'N' page requests. For example, if 'N' is 100 and 5 of the last 100 page requests failed, page availability is shown as 95%.

Table 27–3 (Cont.) Recent Page Request Metrics

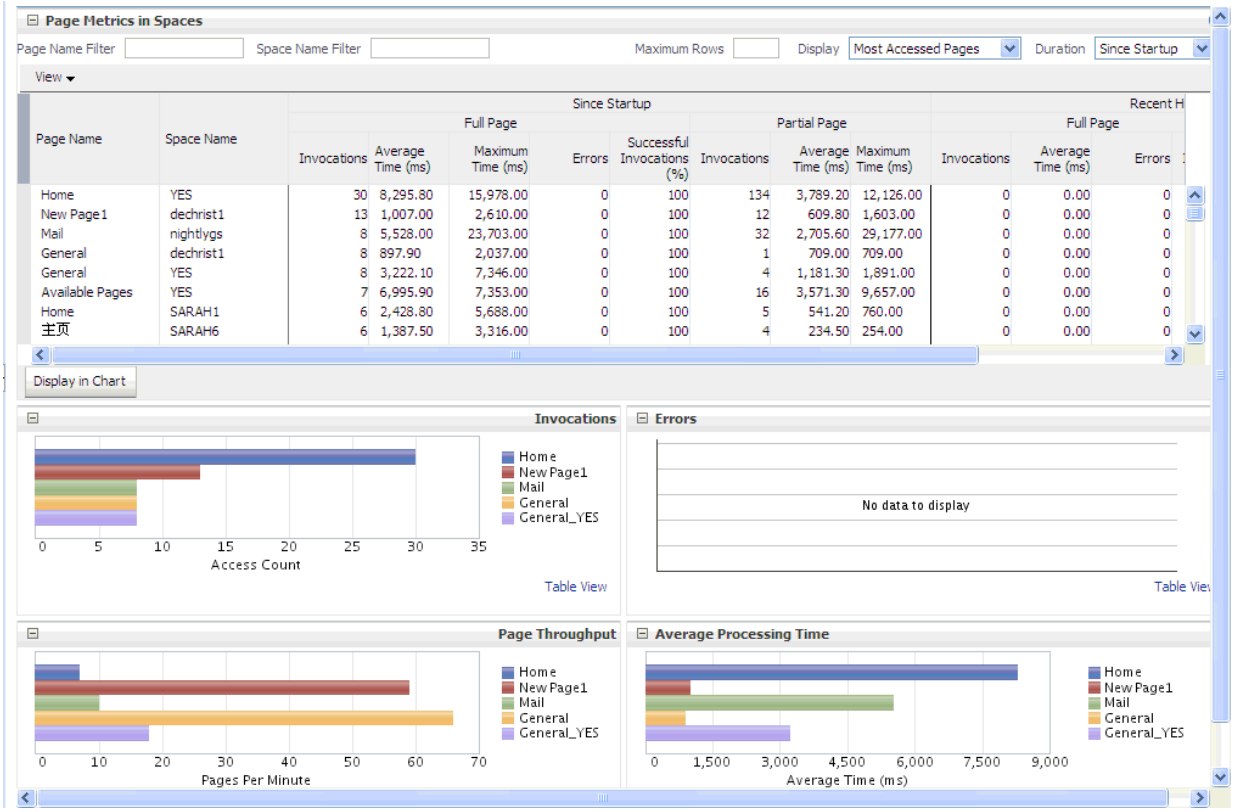
Metric	Description
Performance	Indicates page performance over the last 'N' page requests: ■ **Green** - Indicates acceptable page response times, that is, the time taken to respond is less than a predefined threshold. ■ **Red** - Indicates page responses exceeding the limit set. For example, if your installation specifies the page response threshold to be 3, 000 ms, responses longer then 3, 000 ms trigger a warning message and an "out-of-bounds" condition is logged. Out-of-the-box, the page response threshold is 10, 000ms. Look at the **Time** column in the table below. Responses that exceed the threshold appear in orange. Click the **Sort Descending** arrow to identify the slowest pages. Open and examine slow pages to assess whether there is scope to improve page performance either by redesigning the page or modifying/removing page content. ■ **%** - Percentage of page requests that responded within the time limit specified. The percentage is calculated using information from the last 'N' page requests. For example, if 'N' is 100, and 10 of the last 100 page requests exceeded the page response threshold, page performance is shown as 90%.
Date Time	Date and time page requested.
Page Name	Name of the page requested.
Space Name	(WebCenter Portal only) Name of the portal (previously referred to as a *space*) in which the page is stored.
Partial Page Refresh	Indicates whether the page request refreshed the whole page (`false`) or a part of the page (`true`).
Status	Indicates whether the page request was successful (Success) or failed (Failure). **Failure** displays in orange text.
Time (ms)	Time taken to refresh the page (full or partial), in milliseconds. If the time exceeds the predefined page response threshold, the value displays in "orange".

27.1.5.3 Overall Page Metrics

Historical performance metrics associated with page activity are also available as shown in Figure 27–5 and described in Table 27–4. This page displays metrics for both full and partial page requests and you can filter the data displayed to suit your requirements.

> **Note:** To access these metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

Figure 27–5 Overall Page Request Metrics



The table at the top of this page summarizes the status and performance of individual pages. Use the table to quickly see which pages are available, and to review their individual and relative performances.

Statistics become available when a page is created and are updated every time someone accesses and uses the page.

Note: (WebCenter Portal only) Metrics for pages in the Home portal are not included.

Table 27–4 Page Request Metrics - Full Page and Partial Page

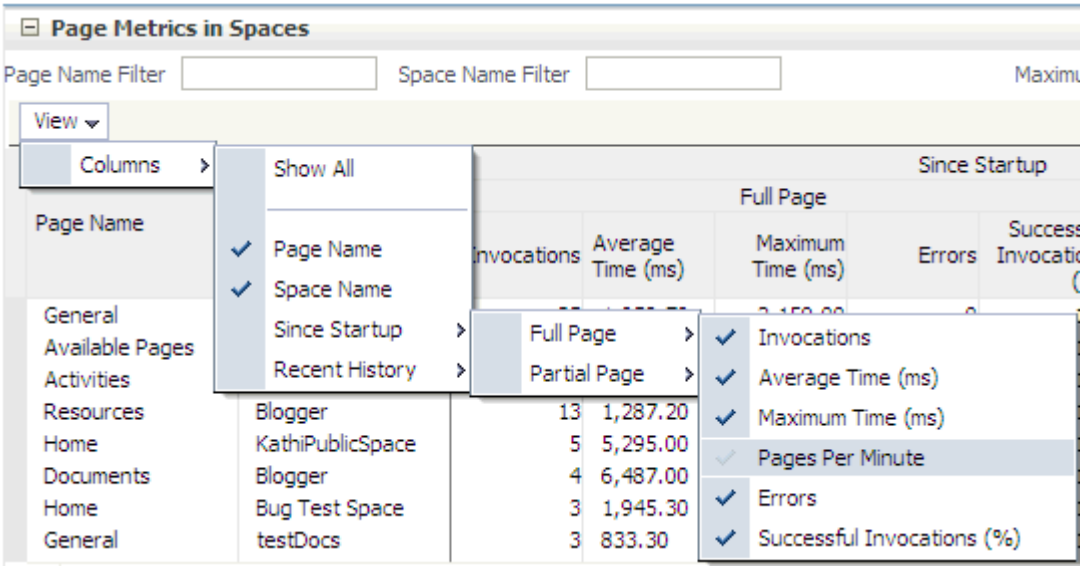
Field	Description
Display Options	Filter the data displayed in the table: ■ **Page Name Filter** - Enter a full or partial search term, then click the **Refresh** icon to refresh the list with all pages for which a match is found in the page name. To display all pages, clear the search term and click **Refresh** again. ■ **Space Name Filter** - (WebCenter Portal only) Enter a full or partial search term, then click the **Refresh** icon to refresh the list with all pages for which a match is found in the portal's display name. To display page metrics from all portals (previously referred to as *spaces*), clear the search term and click **Refresh** again. ■ **Maximum Rows** - Restrict the total number of pages displayed in the table. ■ **Display** - Display metrics for the most popular pages, the slowest pages, or the pages experiencing the most errors. Depending on you selection, the table orders pages by: - Number of Invocations (Most Accessed Pages) - Average Page Processing Time (Slowest Pages) - Number of Errors (Pages with Most Errors) ■ **Duration** - Display metric information collected since startup or in the last 15 minutes (Recent History). The top five pages display in the chart.
Page Name	Names of pages that match your filter criteria (if any). If you do not specify filter criteria, all the pages are listed.
Space Name	(WebCenter Portal only) Names of portals (previously referred to as *spaces*) that match your filter criteria (if any). If you do not specify filter criteria, pages from all portals are listed.
Invocations	Total number of page invocations per minute (full or partial): - Since Startup - Recent History
Average Time (ms)	Average time (in ms) to display the page (full or partial): - Since Startup - Recent History
Maximum Time (ms)	Maximum time taken to display a page (full or partial):
Errors (Only for full page)	Number of errors that occurred for a page per minute.
Successful Invocations (Only for full page)	Percentage of page invocations that succeeded: - Since Startup - Recent History If Successful Invocations (%) is below 100%, check the diagnostic logs to establish why page requests are failing. See Section 28.2, "Viewing and Configuring Log Information."

Table 27–4 (Cont.) Page Request Metrics - Full Page and Partial Page

Field	Description
Pages per Minute	Number of times the page is accessed per minute, also referred to as page throughput: - Since Startup - Recent History

By default, the **Pages Per Minute** for full page metrics is hidden. To show this metric, go to the **View** menu > **Columns** > **Since Startup**/**Recent History** > **Full Page** > **Pages Per Minute** (Figure 27–6). Similarly, to hide columns that are not required, deselect the column names.

Figure 27–6 Pages Per Minute Option in the View Menu



Overall Page Request Metrics - Graphs

Use the graphs below the table to see, at a glance:

- **Invocations** - Graph showing the most popular or least used pages, that is, pages recording the most or least invocations.
- **Page Throughput** - Graph showing the average number of pages accessed per minute. Use this graph to identify pages with high (or low) hit rates.
- **Errors** - (WebCenter Portal only) Graph showing the number of errors. Use this graph to compare error rates.
- **Average Processing Time** - (WebCenter Portal only) Graph showing the average page response time (in milliseconds). Use this graph to identify pages with the best (or worst) performance.
- **Full Page Average Processing Time** - (Portal Framework applications only) Graph showing the average full page response time (in milliseconds). Use this graph to identify pages with the best (or worst) performance.
- **Partial Page Average Processing Time** - (Portal Framework applications only) Graph showing the average partial page response time (in milliseconds). Use this graph to identify pages with the best (or worst) page performance.

To compare a different set of pages:

- Specify the appropriate filtering criteria in the **Page Name Filter**.
- Select one or more pages in the table, and then click **Display in Chart**.

27.1.6 Understanding Document Metrics

Recent document download availability and performance metrics are summarized on the home page for WebCenter Portal or your Portal Framework application (Figure 27–7 and Table 27–5). The document availability/performance charts show at a glance if document download requests are slower than expected or failing

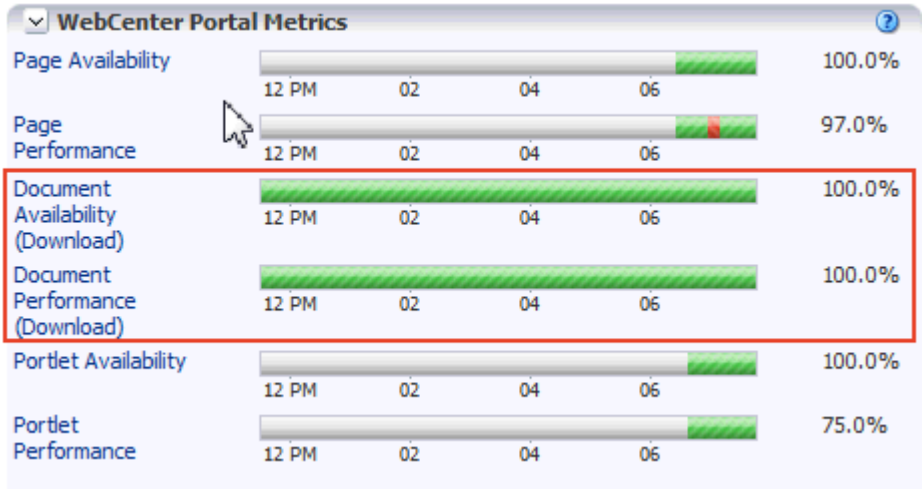
> **Note:** To access the home page, see Section 6.2, "Navigating to the Home Page for WebCenter Portal" or Section 6.3, "Navigating to the Home Page for Portal Framework Applications."

The **Document Download Availability** and **Document Download Performance** charts report availability and performance over the last 'N' document download requests (by default, 'N' is 100). The time range starts with the earliest page/document/portlet request time and ends with the current time. See Section 27.3.6, "Configuring the Number of Samples Used to Calculate Key Performance Metrics."

The % value on the right shows the percentage of document downloads that performed within set thresholds. The percentage is calculated using information from the last 'N' document download requests. For example, if 'N' is 100, and if 3 of the last 100 document download requests are "out-of-bounds", document download performance is shown as 97%. For more information, see Table 27–5.

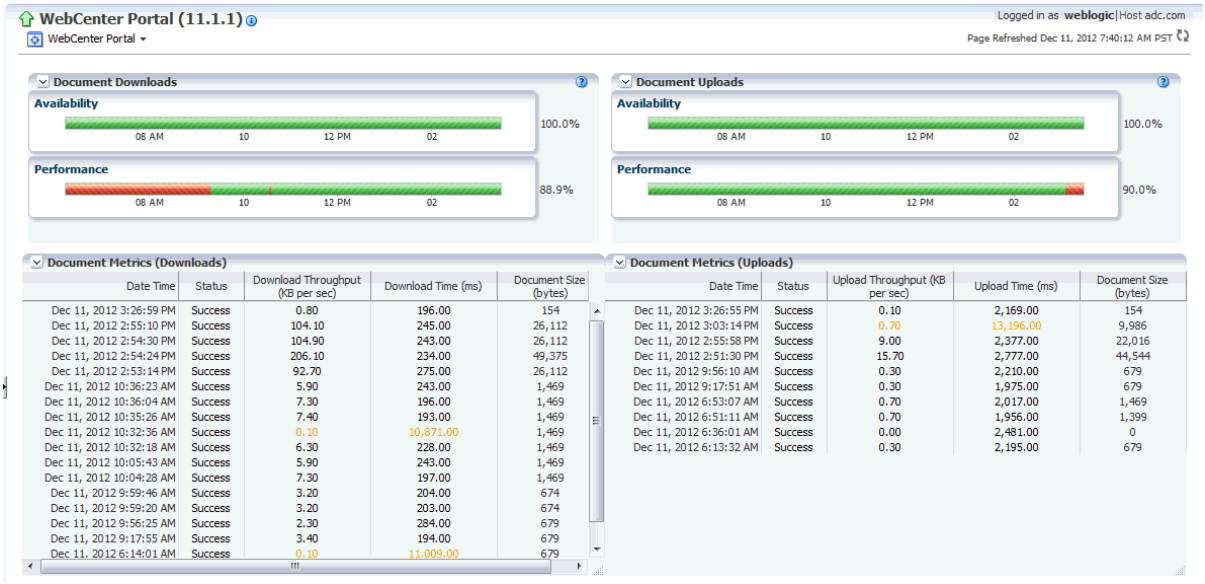
The bar chart status (green/red) does not change over time until the status changes, so the % performance value and the visual green/red ratio do not always match up. An explanation for this is provided in Section 27.1.5.2, "Recent Page Metrics" and the same applies to the document charts.

Figure 27–7 Recent Document Metric Summary on the WebCenter Portal Home Page



If the chart indicates issues or incidents, click the **Document Availability** or **Document Performance** link to navigate to more detailed information to diagnose the issue further (Figure 27–8 and Table 27–5).

Figure 27–8 Recent Document Metrics Page



Use the information on this page to troubleshoot recent, document-related performance issues. The document availability/performance charts at the top of the page indicate "red" if document upload or download requests are slower than expected or failing.

Note: Out-of-the-box, the download threshold is 500ms and the upload threshold is 3, 000ms. Documents taking longer than this to download/upload show "red" in the chart *if* the download/upload throughput rate is less than 1024 or 180 KB/second respectively. If these thresholds are not suitable for your installation you can change the threshold value. See Section 27.3.4, "Configuring Thresholds for Document Upload/Download Metrics."

The charts report availability/performance checks results over the last 'N' document download/upload requests. The time range starts with the earliest document request time and ends with the time of the last document request.

To diagnose document download/upload issues, refer to the advice in "*Step 4. Monitor document uploads and downloads*" in Table 27–2, " Analyzing System Health - Step by Step".

Table 27–5 Recent Document Metrics

Metric	Description
Availability	Indicates document availability over the last 'N' document download/upload requests: ■ **Green** - Indicates successful document downloads/uploads. ■ **Red** - Indicates that a failure occurred during a document download/upload. Look at the **Status** column in the table below to analyze any downloads/uploads that fail. ■ **%** - Percentage of document downloads that succeeded. The percentage is calculated using status information from the last 'N' downloads For example, if 'N' is 100 and 2 of the last 100 downloads failed, availability is shown as 98%.
Performance	Indicates document performance over the last 'N' document download/upload requests: ■ **Green** - Indicates acceptable document performance: *- Documents download/upload within the time threshold specified.* *- Documents exceed the document download/upload throughput threshold (regardless of the overall download time).* ■ **Red** - Indicates document performance is less than the limit set. For example, if your installation specifies an acceptable document download rate to be 3 KB/second and an acceptable download time to be 500ms, any download taking longer than 500ms at a rate less than 3 KB/second triggers a warning message and an "out-of-bounds" condition is logged. Default alert thresholds for document download/upload times (msec) and throughput (KB/second) are provided out-of-the-box. You can fine tune both thresholds to suit your installation. For details, see Section 27.3.4, "Configuring Thresholds for Document Upload/Download Metrics." Look at the **Throughput** column in the table below. Documents download/upload rates lower than the threshold appear in orange. Download/upload times that exceed the threshold display in orange too. Click **Sort Ascending** (Throughput columns) to identify the slowest rates and **Sort Descending** (Time columns) to find the longest times. ■ **%** - Percentage of documents that download/upload at an acceptable rate. The percentage is calculated using information from the last 'N' downloads/uploads. For example, if 'N' is 100, and 10 of the last 100 downloads failed to meet the predefined download threshold, document download performance is shown as 90%.
Date Time	Date and time a document was uploaded/downloaded.
Status	Indicates whether the document upload/download operation was successful (Success) or failed (Failure). **Failure** displays in orange text.
Download/Upload Throughput (KB per second)	Amount of document data downloaded/uploaded per second. If the throughput fails to meet a predefined response, the value displays in "orange".

Table 27–5 (Cont.) Recent Document Metrics

Metric	Description
Download/Upload Time (ms)	Time taken to download/upload the document.
Document Size (bytes)	Size of the document (in bytes). Use the Sort Ascending/Sort Descending icons to analyze the performance by document size.

27.1.7 Understanding Portlet Producer Metrics

You can monitor the availability and performance of all the portlets and portlet producers used by WebCenter Portal or Portal Framework applications through Fusion Middleware Control. You can monitor recent and historical (overall) portlet data. The following topics describe the metrics that are available:

- Section 27.1.7.1, "Recent Portlet Metrics"
- Section 27.1.7.2, "Overall Portlet Producer Metrics"
- Section 27.1.7.3, "Overall Portlet Metrics"

27.1.7.1 Recent Portlet Metrics

Recent portlet availability and performance metrics are summarized on the home page for WebCenter Portal or your Portal Framework application (Figure 27–9 and Table 27–6). The portlet availability/performance charts show at a glance if portlet requests are slower than expected or failing.

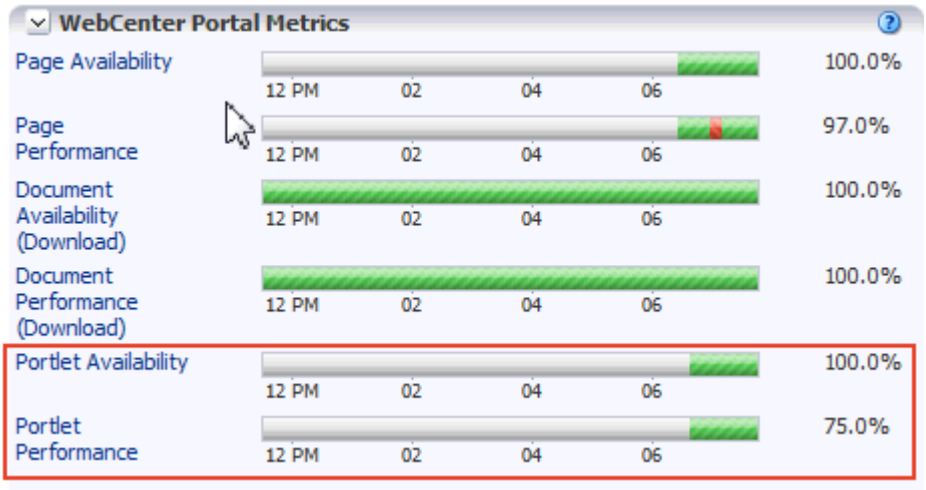
> **Note:** To access the home page, see Section 6.2, "Navigating to the Home Page for WebCenter Portal" or Section 6.3, "Navigating to the Home Page for Portal Framework Applications."

The **Portlet Availability** and **Portlet Performance** charts report availability and performance over the last 'N' portlet requests (by default, 'N' is 100). The time range starts with the earliest page/document/portlet request time and ends with the current time. See Section 27.3.6, "Configuring the Number of Samples Used to Calculate Key Performance Metrics."

The % value on the right shows the percentage of portlet requests that responded within a specific time limit. The percentage is calculated using information from the last 'N' portlet requests. For example, if 'N' is 100, and if 25 of the last 100 portlet requests exceeded the portlet response threshold, portlet performance is shown as 75%. For more information, see Table 27–6.

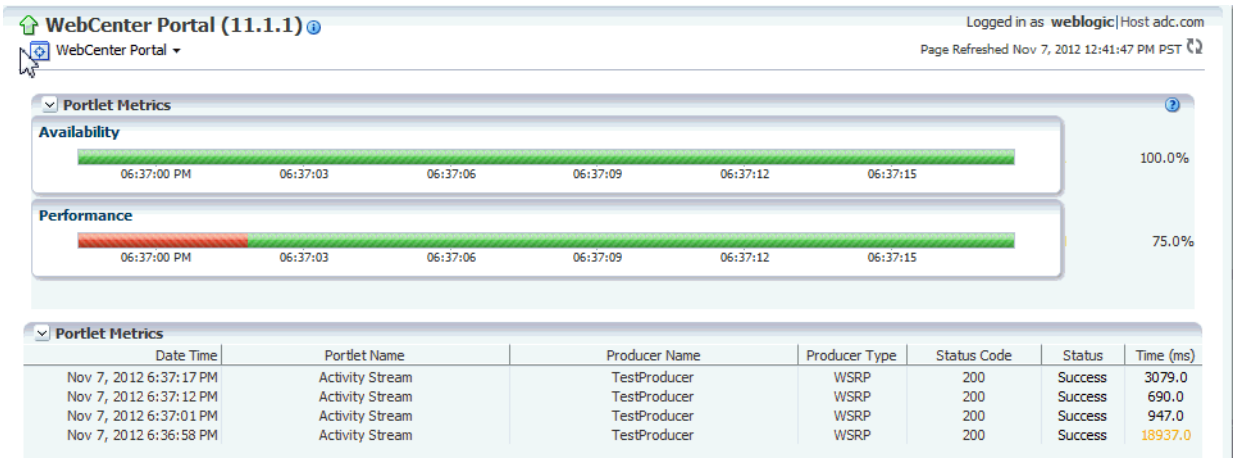
The bar chart status (green/red) does not change over time until the status changes, so the % performance value and the visual green/red ratio do not always match up. An explanation for this is provided in Section 27.1.5.2, "Recent Page Metrics" and the same applies to the portlet charts.

Figure 27–9 Recent Portlet Metric Summary on the WebCenter Portal Home Page



If the chart indicates issues or incidents, click the **Portlet Availability** or **Portlet Performance** link navigate to more detailed information to diagnose the issue further (Figure 27–10 and Table 27–6).

Figure 27–10 Recent Portlet Metrics



Use the information on this page to troubleshoot recent portlet performance issues. The portlet availability/performance charts at the top of the page show "red" if portlet requests are slower than expected or failing.

> **Note:** Out-of-the-box, the portlet response threshold is 10, 000ms so portlets taking longer than 10, 000ms to respond show "red" in the chart. If this threshold is not suitable for your installation you can change the threshold value. For more information, see "Section 27.3, "Customizing Key Performance Metric Thresholds and Collection."

The charts report availability/performance over the last 'N' portlet requests. The time range starts with the earliest portlet request time and ends with the time of the last portlet request.

Use the information in the table to identify slow portlets. You can determine the name of the portlet and the producer to which the portlets belongs.

To diagnose portlet issues, refer to the advice in "*Step 5. Monitor portlet performance*" in Table 27–2, " Analyzing System Health - Step by Step".

Table 27–6 Recent Portlet Metrics

Portlet Availability	Indicates portlet availability over the last 'N' portlet requests: ■ **Green** - Indicates successful portlet requests. ■ **Red** - Indicates that a failure occurred during a portlet request. Look at the **Status** column in the table below to identify any portlet requests that fail. ■ **%** - Percentage of portlet requests that succeeded. The percentage is calculated using status information from the last 'N' portlet requests. For example, if 'N' is 100 and 5 of the last 100 portlet requests failed, portlet availability is shown as 95%.
Portlet Performance	Indicates portlet performance over the last 'N' portlet requests: ■ **Green** - Indicates acceptable portlet response times, that is, the time taken to respond is less than a predefined threshold. ■ **Red** - Indicates portlet responses exceeding the limit set. For example, if your installation specifies the portlet response threshold to be 60 ms, responses longer then 60 ms trigger a warning message and an "out-of-bounds" condition is logged. Out-of-the-box, the portlet response threshold is 10,000ms. Look at the **Time** column in the table below. Responses that exceed the threshold appear in orange. Click the **Sort Descending** arrow to identify the slowest portlets. Once you have the portlet's name, you can examine the portlet to assess how they might be modified to improve efficiency. ■ **%** - Percentage of portlet requests that responded within the time limit specified. The percentage is calculated using information from the last 'N' portlet requests. For example, 'N' is 100, and 10 of the last 100 portlet requests exceeded the portlet response threshold, portlet performance is shown as 90%.
Date Time	Date and time of the portlet request.
Portlet Name	Name of the portlet requested.

27.1.7.2 Overall Portlet Producer Metrics

Historical performance metrics are also available for portlet producers used by WebCenter Portal or Portal Framework applications, as shown in Figure 27–11. The information displayed on this page is described in the following tables:

- Table 27–7, " Portlet Producers - Summary"
- Table 27–8, " Portlet Producer - Detail"

Note: To access these metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

Figure 27–11 Portlet Producer Metrics

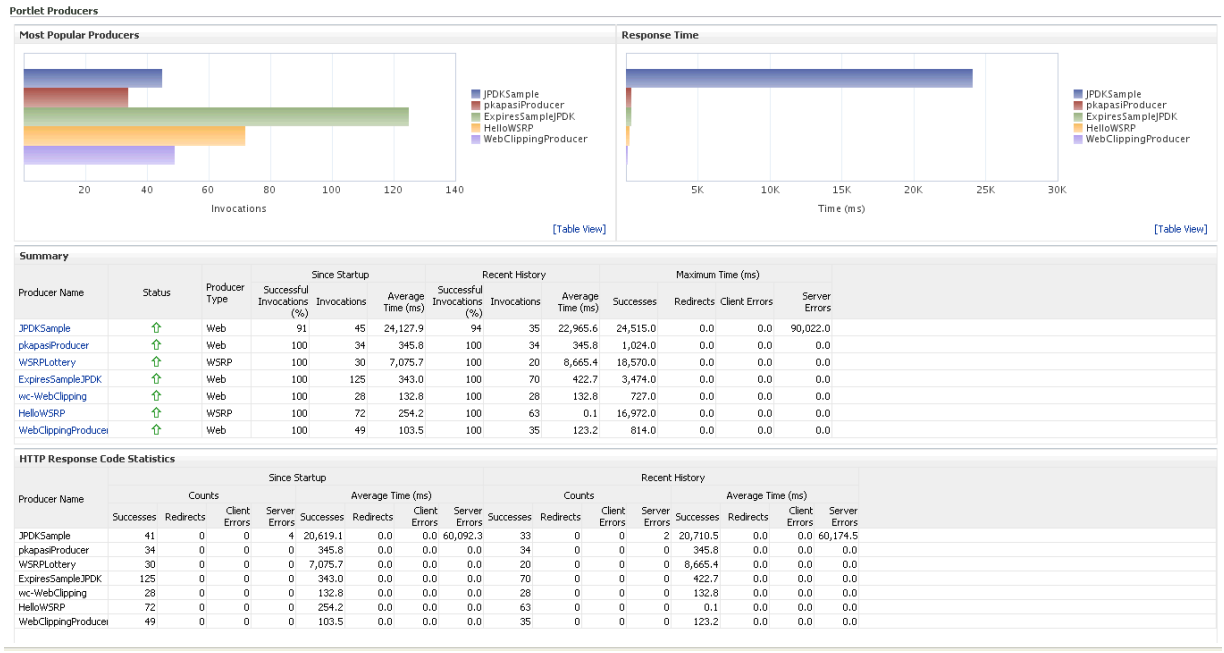


Table 27–7 Portlet Producers - Summary

Metric	Description
Status	The current status of portlet producers used in the application: ■ **Up** (Green Up Arrow) - Indicates that all portlet producers are up and running. ■ **Down** (Red Down Arrow) - Indicates that the one or more portlet producers are currently unavailable. A producer instance might be down, or there could be some network connectivity issues. ■ **Unknown (Clock)** - Unable to query the status of the portlet producers for some reason. Maybe the managed server is down or the node cannot be reached due to a network issue. To diagnose further, review the Admin Server log, and the managed server logs.
Successful Invocations (%)	The percentage of portlet producer invocations that succeeded: - Since Startup - Recent History Any request that fails will impact availability. This includes application-related failures such as timeouts and internal errors, and also client/server failures such as requests returned with response codes HTTP4xx or HTTP5xx, responses with a bad content type, and SOAP faults, where applicable. If Successful Invocations (%) is below 100%, check the diagnostic logs to establish why service requests are failing. See Section 28.2, "Viewing and Configuring Log Information."

Table 27–7 (Cont.) Portlet Producers - Summary

Metric	Description
Invocations	The number of portlet producer invocations per minute: - Since Startup - Recent History This metric measures each application-related portlet request and therefore, due to cache hits, errors, or timeouts on the application, this total may be higher than the number of actual HTTP requests made to the producer server.
Average Time (ms)	The average time taken to make a portlet request, regardless of the result: - Since Startup - Recent History

Table 27–8 Portlet Producer - Detail

Metric	Description
Most Popular Producers	The number of invocations per producer (displayed on a chart). The highest value on the chart indicates which portlet producer is used the most. The lowest value indicates which portlet producer is used the least.
Response Time	The average time each portlet producer takes to process producer requests since WebCenter Portal or your Portal Framework application started up (displayed on a chart). The highest value on the chart indicates the worst performing portlet producer. The lowest value indicates which portlet producer is performing the best.
Producer Name	The name of the portlet producer being monitored. Click the name of a portlet producer to pop up more detailed information about each portlet that the application uses. See Table 27–10, " Portlet - Detail".
Status	The current status of each portlet producer: ■ **Up** (Green Up Arrow) - Indicates that the portlet producer is up and running. ■ **Down** (Red Down Arrow) - Indicates that the portlet producer is currently unavailable. The producer instance might be down, or there could be some network connectivity issues. ■ **Unknown (Clock)** - Unable to query the status of portlet producer for some reason.
Producer Type	The portlet producer type: Web or WSRP ■ Web portlet producer - Oracle PDK Java producer deployed to a J2EE application server, which is often remote and communicates through Simple Object Access Protocol (SOAP) over HTTP. ■ WSRP portlet producer - Web Services for Remote Portlets (WSRP) is a Web services standard that allows interoperability between a standards enabled container and any WSRP application.

Table 27–8 (Cont.) Portlet Producer - Detail

Metric	Description
Successful Invocations (%)	The percentage of producer invocations that succeeded: - Since Startup - Recent History
Invocations	The number of invocations, per producer: - Since Startup - Recent History By sorting the table on this column, you can find the most frequently accessed portlet producer in WebCenter Portal or your Portal Framework application.
Average Time (ms)	The average time taken to make a portlet request, regardless of the result: - Since Startup - Recent History Use this metric to detect non-functional portlet producers. If you use this metric with the Invocations metric, then you can prioritize which producer to focus on.
Maximum Time (ms)	The maximum time taken to process producer requests: - Successes - HTTP200xx response code - Re-directs - HTTP300xx response code - Client Errors - HTTP400xx response code - Server Errors - HTTP500xx response code

27.1.7.3 Overall Portlet Metrics

Historical performance metrics are available for individual portlets used by WebCenter Portal or a Portal Framework application, as shown in Figure 27–12. The information displayed on this page is described in the following tables:

- Table 27–9, " Portlets - Summary"
- Table 27–10, " Portlet - Detail"
- Table 27–11, " Portlet - HTTP Response Code Statistics"
- Table 27–12, " HTTP Response Codes"

> **Note:** To access these metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

Figure 27–12 Portlet Metrics

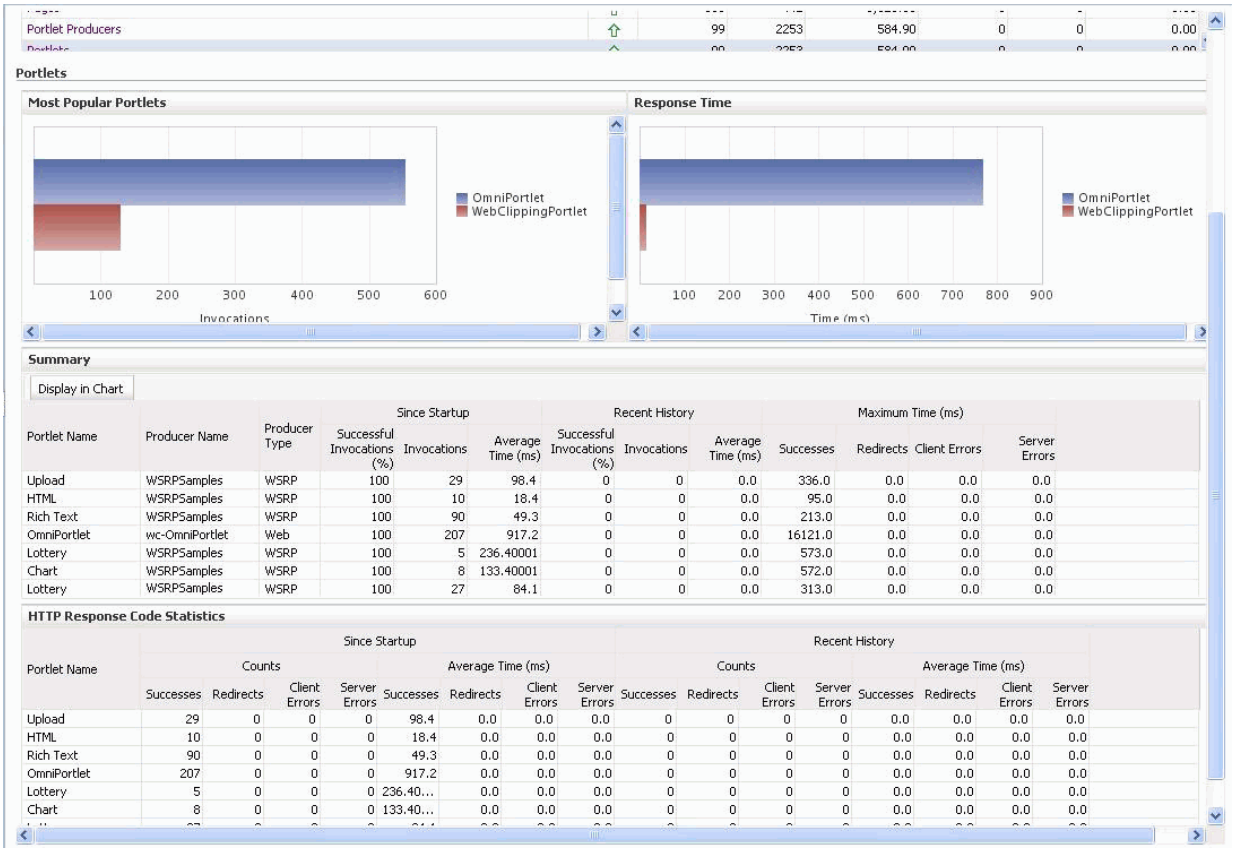


Table 27–9 Portlets - Summary

Metric	Description
Status	The current status of portlets used in WebCenter Portal or your Portal Framework application: ▪ **Up** (Green Up Arrow) - Indicates that all portlets are up and running. ▪ **Down** (Red Down Arrow) - Indicates that the one or more portlets are currently unavailable. A producer instance might be down, or there could be some network connectivity issues. For other causes, see Section 27.1.12.3.13, "Portlets and Producers - Issues and Actions." ▪ **Unknown (Clock)** - Unable to query the status of portlets for some reason. Maybe the managed server is down or the node cannot be reached due to a network issue. To diagnose further, review the Admin Server log, and the managed server logs.

Table 27–9 (Cont.) Portlets - Summary

Metric	Description
Successful Invocations (%)	The percentage of portlet invocations that succeeded: - Since Startup - Recent History Any request that fails will impact availability. This includes application-related failures such as timeouts and internal errors, and also client/server errors. If Successful Invocations (%) is below 100%, check the diagnostic logs to establish why service requests are failing. See Section 28.2, "Viewing and Configuring Log Information."
Invocations	The number of portlet invocations per minute: - Since Startup - Recent History This metric measures each application-related portlet request and therefore, due to cache hits, errors, or timeouts on the application, this total may be higher than the number of actual HTTP requests made to the portlet producer.
Average Time (ms)	The average time taken to process operations associated with portlets, regardless of the result: - Since Startup - Recent History

Table 27–10 Portlet - Detail

Metric	Description
Most Popular Portlets	The number of invocations per portlet (displayed on a chart). The highest value on the chart indicates which portlet is used the most. The lowest value indicates which portlet is used the least.
Response Time	The average time each portlet takes to process requests since WebCenter Portal or your Portal Framework application started up (displayed on a chart). The highest value on the chart indicates the worst performing portlet. The lowest value indicates which portlet is performing the best.
Portlet Name	The name of the portlet being monitored.
Status	The current status of each portlet: ■ **Up** (Green Up Arrow) - Indicates that the portlet is up and running. ■ **Down** (Red Down Arrow) - Indicates that the portlet is currently unavailable. The producer instance might be down, or there could be some network connectivity issues.
Producer Name	The name of the portlet producer through which the portlet is accessed.

Table 27–10 (Cont.) Portlet - Detail

Metric	Description
Producer Type	The portlet producer type: Web or WSRP ■ Web portlet producer - Oracle PDK Java producer deployed to a J2EE application server, which is often remote and communicates through Simple Object Access Protocol (SOAP) over HTTP. ■ WSRP portlet producer - Web Services for Remote Portlets (WSRP) is a Web services standard that allows interoperability between a standards enabled container and any WSRP application.
Successful Invocations (%)	The percentage of portlet invocations that succeeded: - Since Startup - Recent History If Successful Invocations (%) is below 100%, check the diagnostic logs to establish why service requests are failing. See Section 28.2, "Viewing and Configuring Log Information."
Invocations	The number of invocations, per portlet: - Since Startup - Recent History By sorting the table on this column, you can find the most frequently accessed portlet in WebCenter Portal or your Portal Framework application.
Average Time (ms)	The average time each portlet takes to process requests, regardless of the result: - Since Startup - Recent History Use this metric to detect non-performant portlets. If you use this metric with the Invocations metric, then you can prioritize which portlet to focus on.
Maximum Time (ms)	The maximum time taken to process portlet requests: - Successes - HTTP200xx - Redirects - HTTP300xx - Client Errors - HTTP400xx - Server Errors - HTTP500xx The breakdown of performance statistics by HTTP response code can help you identify which factors are driving up the total average response time. For example, failures due to portlet producer timeouts would adversely affect the total average response time.

Table 27–11 Portlet - HTTP Response Code Statistics

Metric	Description
Portlet Name	The name of the portlet being monitored.

Table 27–11 (Cont.) Portlet - HTTP Response Code Statistics

Metric	Description
Invocations Count	The number of invocations, by type (HTTP response code):
- Successes	- Since Startup
- Redirects	- Recent History
- Client Errors	See Table 27–12, " HTTP Response Codes".
- Server Errors	
Average Time (ms)	The average time each portlet takes to process requests:
- Successes	- Since Startup
- Redirects	- Recent History
- Client Errors - Server Errors	Use this metric to detect non-functional portlets. If you use this metric with the Invocations metric, then you can prioritize which portlet to focus on.

Table 27–12 HTTP Response Codes

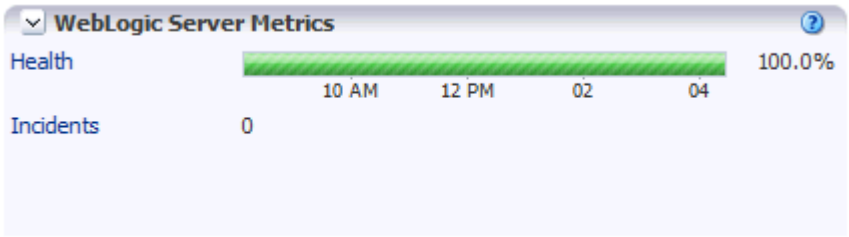
HTTP Response and Error Code	Description
200 -Successful Requests	Portlet requests that return any HTTP2xx response code, or which were successful without requiring an HTTP request to the remote producer, for example, a cache hit.
300 -Unresolved Redirections	Portlet requests that return any HTTP3xx response code.
400 -Unsuccessful Request Incomplete	Portlet requests that return any HTTP4xx response code.
500 -Unsuccessful Server Errors	Portlet requests that failed for any reason, including requests that return HTTP5xx response codes, or which failed due to a application-related error, timeout, bad content type response, or SOAP fault.

27.1.8 Understanding WebLogic Server Metrics

Recent WebLogic Server performance is summarized on the home page for WebCenter Portal or your Portal Framework application (Figure 27–13 and Table 27–13). If the chart indicates issues or incidents, you can navigate to more detailed information to diagnose the issue further.

> **Note:** To access the home page, see Section 6.2, "Navigating to the Home Page for WebCenter Portal" or Section 6.3, "Navigating to the Home Page for Portal Framework Applications."

Figure 27–13 Recent WebLogic Server Metric Summary on the Home Page



The charts report results from the last WebLogic Server 100 health checks. By default, metrics are recorded every five minutes so data collected over the last 8 hours can display here. If the server started up recently, the chart displays data from the time the server started to the current time.

> **Note:** If required, you can customize the metric collection frequency to better suit your installation. For details, see Section 27.3, "Customizing Key Performance Metric Thresholds and Collection."

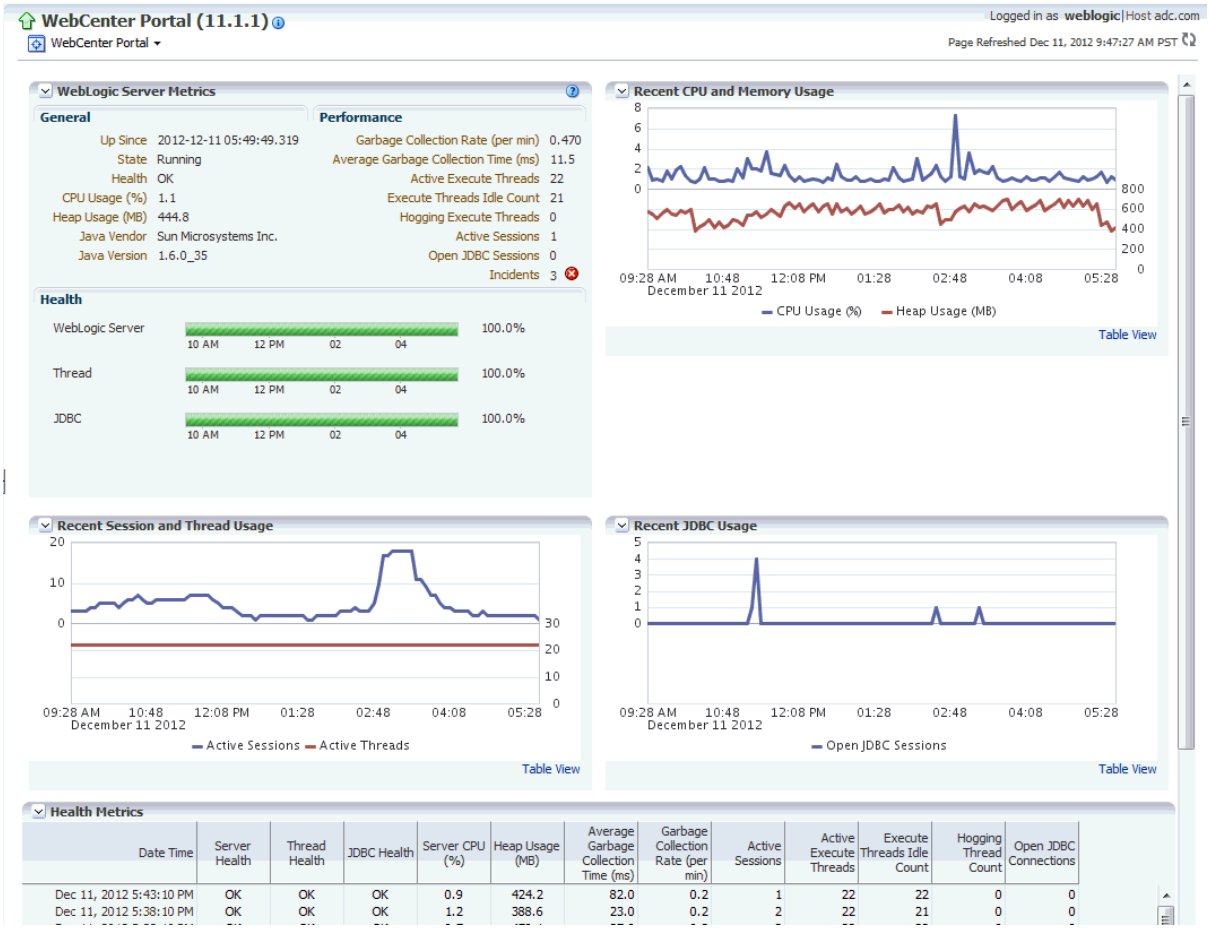
Table 27–13 Recent WebLogic Server Metrics on the Home Page

Metric	Description
Health	Summarizes recent WebLogic Server health as reported by the Oracle WebLogic Server self-health monitoring feature. This metric considers recent server health, thread health, and JDBC health: ■ **Green** - Indicates successful WebLogic Server health checks. ■ **Red** - Indicates that an incident occurred during a WebLogic Server health check. Click **Health** to identify health checks that fail (do not report OK). See Figure 27–14. ■ % - Percentage of WebLogic Server health checks that succeeded. By default, the percentage is calculated using status information from the last 100 health checks. For example, if 5 of the last 100 health checks fail (do not report OK), Health is shown as 95%.
Incidents	Number of times WebLogic Server metrics exceed threshold settings (that is, metrics such as CPU usage, memory usage, thread count, number of JDBC connections, session metrics, and so on). For example, if the metric data set contains 2 incidents where thread count exceeded the predefined threshold and the number of JDBC connections exceeded the threshold limit 3 times, then the number of incidents displayed is 5. When the number of incidents is greater than 0, an icon with a red cross displays. Click the **Incidents** link to drill down to the Recent WebLogic Server Metrics Page (Figure 27–13) and examine the Health Metrics table to diagnose the incidents further.

You can click **Health** or **Incidents** to drill down to the Recent WebLogic Server Metrics Page (Figure 27–13). The metrics displayed on this page are described in the following topics:

- WebLogic Server Metrics Section
- Recent CPU and Memory Usage Section
- Recent Session and Thread Usage Section
- Recent JDBC Usage Section
- Health Metrics Section

Figure 27–14 Recent WebLogic Server Metrics Page



WebLogic Server Metrics Section

Metric	Description
General	
Up Since	Date and time the server last started up.
State	Current lifecycle state of this server. For example, a server can be in a RUNNING state in which it can receive and process requests or in an ADMIN state in which it can receive only administrative requests. For more information, see the "Understanding Server Life Cycle" section in *Oracle Fusion Middleware Managing Server Startup and Shutdown for Oracle WebLogic Server*.
Health	Health status of the server, as reported by the Oracle WebLogic Server self-health monitoring feature. For example, the server can report if it is overloaded by too many requests, if it needs more memory resources, or if it will soon fail for other reasons. For more information, see the "Configure health monitoring," section in the Oracle WebLogic Server Administration Console online help.

Metric	Description
CPU Usage (%)	Percentage of the CPU currently in use by the Java Virtual Machine (JVM). This includes the load that the JVM is placing on all processors in the host computer. For example, if the host uses multiple processors, the value represents a snapshot of the average load on all the processors.
Heap Usage (MB)	Size of the memory heap currently in use by the Java Virtual Machine (JVM), in megabytes.
Java Vendor	Name of the company that provided the current Java Development Kit (JDK) on which the server is running.
Java Version	Version of the JDK on which the current server is running.
Performance	
Garbage Collection Rate (per min)	Rate (per minute) at which the Java Virtual Machine (JVM) is invoking its garbage-collection routine. By default, this metric shows the rate recorded in the last five minutes. See Section 27.3.5, "Configuring the Frequency of WebLogic Server Health Checks."
Average Garbage Collection Time (ms)	Average length of time (ms) the Java Virtual Machine spent in each run of garbage collection. The average shown is for the last five minutes. By default, this metric shows the average over the last five minutes. See Section 27.3.5, "Configuring the Frequency of WebLogic Server Health Checks."
Active Execute Threads	Number of active execute threads in the pool.
Execute Threads Idle Count	Number of idle threads in the pool. This count does not include standby threads or stuck threads. The count indicates threads that are ready to pick up new work when it arrives.
Hogging Execute Threads	Number of threads that are being held by a request right now. These threads will either be declared as stuck after a configured timeout or return to the pool. The self-tuning mechanism backfills if necessary.
Active Sessions	Number of active sessions for the application.
Open JDBC Sessions	Number of JDBC connections currently open.
Incidents	Number of times WebLogic Server metrics exceed threshold settings (that is, metrics such as CPU usage, memory usage, thread count, number of JDBC connections, session metrics, and so on). For example, if the metric data set contains 2 incidents where thread count exceeded the predefined threshold and the number of JDBC connections exceeded the threshold limit 3 times, then the number of incidents displayed is 5. When the number of incidents is greater than 0, an icon with a red cross displays.

Metric	Description
Health	Summarizes recent health status, as reported by the Oracle WebLogic Server self-health monitoring feature. The Health charts report results from the last 100 performance checks. By default, metrics are recorded every five minutes so data collected over the last 500 minutes displays. If the server started up recently, the chart displays data from the time the server started to the current time. ■ **Green** - Indicates successful health checks, that is, checks that return "OK". ■ **Red** - Indicates that a health check returned a status other than "OK". For example, if all threads in the default queue become stuck, server health state changes to "CRITICAL". Similarly, if all threads in `weblogic.admin.HTTP`, `weblogic.admin.RMI`, or a user-defined execute queue become stuck, server health state changes to "WARNING". To identify failed health checks, review the Health Metrics Section at the bottom of the page. ■ **%** - Percentage of health checks that succeeded (OK). The percentage is calculated using status information from the last 100 health checks. For example, if 5 of the last 100 thread health checks fail, thread health is shown as 95%.
WebLogic Server	Reports recent WebLogic Server health checks. For example, if 10 out of the last 100 WebLogic Server health checks failed (not "OK"), WebLogic Server health is shown as 90%.
Thread	Reports recent thread health checks. For example, if 10 out of the last 100 WebLogic Server health checks report a thread health status other than "OK", WebLogic Server thread health is shown as 90% Some example thread health failures include: ■ If all threads in the default queue become stuck, server health state changes to "CRITICAL". ■ If all threads in `weblogic.admin.HTTP`, `weblogic.admin.RMI`, or a user-defined execute queue become stuck, server health state changes to "WARNING".
JDBC	Reports recent JDBC health checks. For example, the server can report too many JDBC connection requests. If 10 out of the last 100 WebLogic Server health checks report a JDBC health status other than "OK", WebLogic Server JDBC health is shown as 90%.

Recent CPU and Memory Usage Section

This graph charts CPU and memory utilization for the Java Virtual machine over the the last 100 health checks.The time range starts with the earliest health check and ends with the time of the last health check.

From this performance graph, you will be able to tell how much of the memory/CPU configured for the virtual machine is actually being used and whether the trend is increasing. This might reveal to you that the applications running inside that virtual machine need more memory than the virtual machine has been assigned and that adding more memory to the virtual machine -- assuming that there is sufficient memory at the host level -- might improve performance. Similarly, you can assess whether additional CPU resources are required.

Metric	Description
CPU Usage (%)	Percentage of the CPU currently in use by the Java Virtual Machine (JVM). This includes the load that the JVM is placing on all processors in the host computer. For example, if the host uses multiple processors, the value represents a snapshot of the average load on all the processors.
Heap Usage (MB)	Size of the memory heap currently in use by the Java Virtual Machine (JVM), in megabytes.

Recent Session and Thread Usage Section

This graph charts the number of active sessions and active threads recorded over the last 100 health checks.The time range starts with the earliest health check and ends with the time of the last health check.

The number of active sessions and threads should rise and fall with the load on your system. If the graph shows a sudden rise or the number of sessions or threads keep increasing, investigate the issue further to understand what triggered the change in behavior.

Metric	Description
Active Sessions	Number of active sessions for the application.
Active Thread	Number of active threads for the application.

Recent JDBC Usage Section

This graph charts the number of open JDBC sessions recorded over the last 100 health checks. The time range starts with the earliest health check and ends with the time of the last health check.

The *Current Active Connection Count* metric across all the data sources belonging to the server are used to calculate the overall open JDBC session count displayed here.

Use this chart to determine the number of JDBC sessions being used and to see whether the system is leaking JDBC resources. You can use the information in this chart to assess whether JDBC configuration or the connection pool size needs to be adjusted.

See Section G.4.2.4, "Verifying Connection Pool Settings."

Health Metrics Section

This table displays data from the last 100 WebLogic Server health metrics collected, as reported by the Oracle WebLogic Server self-health monitoring feature.

Metric	Description
Date Time	Date and time of the WebLogic Server health check.
Server Health	Sever health status, as reported by the Oracle WebLogic Server self-health monitoring feature. Successful health checks return "**OK**". Unsuccessful health checks report various failures, for example, the server can report if it is overloaded by too many requests, if it needs more memory resources, or if it will soon fail for other reasons. For more information, see the "Configure health monitoring" section in the Oracle WebLogic Server Administration Console online help.

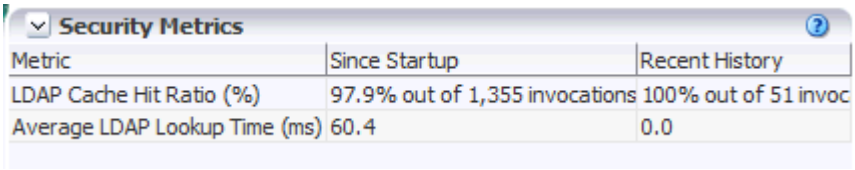
Metric	Description
Thread Health	Thread health status, as reported by the Oracle WebLogic Server self-health monitoring feature. Successful health checks return "**OK**". Unsuccessful thread checks report various failures, for example, if all the threads in the default queue become stuck, server health state changes to "**CRITICAL**". If all threads in `weblogic.admin.HTTP`, `weblogic.admin.RMI`, or a user-defined execute queue become stuck, server health state changes to "**WARNING**". For more information, see the "Configure health monitoring" section in the Oracle WebLogic Server Administration Console online help.
JDBC Health	JDBC health status, as reported by the Oracle WebLogic Server self-health monitoring feature. Successful health checks return "**OK**". Unsuccessful JDBC checks report various failures, for example, if the server reports too many JDBC connection requests or that more memory resources are required, server health state changes to "**WARNING**". For more information, see the "Configure health monitoring" section in the Oracle WebLogic Server Administration Console online help.
Server CPU (%)	If you are using the Oracle JRocket JDK, this metric shows the percentage of the CPU currently in use by the Jave Virtual Machine (JVM). This includes the load that the JVM is placing on all processors in the host computer. For example, if the host uses multiple processors, the value represents a snapshot of the average load on all the processors.
Heap Usage (MB)	Total heap memory (in MB) currently in use by the JVM.
Average Garbage Collection Time (ms)	Average length of time (ms) the Java Virtual Machine spent in each run of garbage collection. The average shown is for the last five minutes. By default, this metric shows the average over the last five minutes. See Section 27.3.5, "Configuring the Frequency of WebLogic Server Health Checks."
Garbage Collection Rate (per min)	Rate (per minute) at which the Java Virtual Machine (JVM) is invoking its garbage-collection routine. By default, this metric shows the rate recorded in the last five minutes. See Section 27.3.5, "Configuring the Frequency of WebLogic Server Health Checks."
Active Sessions	Number of active sessions for the application.
Active Execute Threads	Number of active execute threads in the pool.
Execute Threads Idle Count	Number of idle threads in the pool. This count does not include standby threads or stuck threads. The count indicates threads that are ready to pick up new work when it arrives.
Hogging Thread Count	Number of threads that are being held by a request right now. These threads will either be declared as stuck after a configured timeout or return to the pool. The self-tuning mechanism backfills if necessary.
Open JDBC Connections	Number of JDBC connections currently open.

27.1.9 Understanding Security Metrics

Some key security-related performance metrics are displayed on the home page for WebCenter Portal or your Portal Framework application (Figure 27–15 and Table 27–14).

Note: To access the home page, see Section 6.2, "Navigating to the Home Page for WebCenter Portal" or Section 6.3, "Navigating to the Home Page for Portal Framework Applications."

Figure 27–15 Security Metrics on the Home Page



If you compare **Since Startup** metrics with **Recent History** metrics you can determine whether performance has recently deteriorated, and if so, by how much.

Table 27–14 Security Metrics

Metric	Description
LDAP Cache Hit Ratio (%)	Percentage of LDAP searches that result in a cache hit. WebCenter Portal caches user profiles to improve performance. By default, 1000 profiles are cached and to keep the cache fresh, any cached profile that is not accessed within 60 seconds is thrown out of the cache. After your system has warmed up and the cache populated, this metric should be close to 100%. Under typical conditions, cache hit ratios are above 90%. If the hit ratio is less than 90 percent, consider increasing the length of time profile information is stored in the cache and/or the increase the number of profile objects that can live in the cache using the WLST commands "`setProfileCacheTimeToLive`" and "`setProfileCacheNumberOfObjects`" respectively. For more information, see "Section 16.9, "Configuring Cache Options for the Profile Service." **Note:** The hit ratio is always low after a system restart and gradually rises as users access the WebCenter Portal application.
Average LDAP Lookup Time (ms)	Average time to complete an LDAP search request: - Since Startup - Recent History[1] If LDAP searches are taking too long, its most likely an issue on the LDAP server that is causing slow response times. If you are using Oracle Internet Directory, see the "Oracle Internet Directory Performance Tuning" section in *Oracle Fusion Middleware Performance and Tuning Guide* for advice on how to improve performance and avoid bottlenecks. For other LDAP servers, refer to the appropriate product documentation.

[1] The last 10-15 minutes of data is used to calculate recent performance metrics. For details, see "WebCenter Portal Metric Collection: Recent History and Since Startup".

27.1.10 Understanding Page Response and Load Metrics

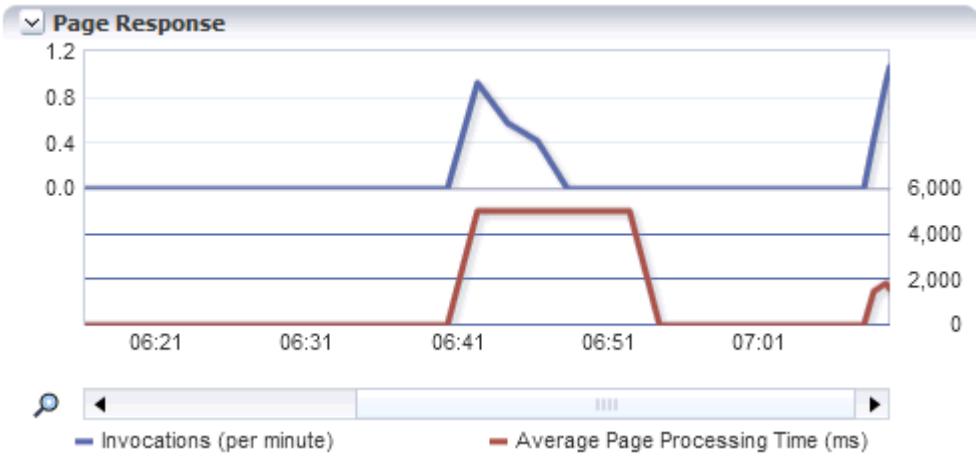
The page response chart on your application's home page (Figure 27–15) shows you how quickly WebLogic Server is responding to page requests and how many requests are being processed (its load).

The average page processing time (in ms) for all portals, is calculated over a 15 minute period. The number of invocations per minute is also displayed to help you determine whether the average page processing time is increasing or decreasing. If slower page processing times are due to a large number of users accessing the system, an increase in invocations per minute will display on the graph. If the number of users has not increased (the invocations per minute graph is not increasing or fluctuating), then slower page processing times are most likely due to machine resource issues or lack of JVM resources (low memory, contention for database connections, and so on).

Click **Table View** to see detailed response and load values, recorded at 5 minute intervals.

> **Note:** To access the home page, see Section 6.2, "Navigating to the Home Page for WebCenter Portal" or Section 6.3, "Navigating to the Home Page for Portal Framework Applications."

Figure 27–16 Page Response Metrics on the Home Page



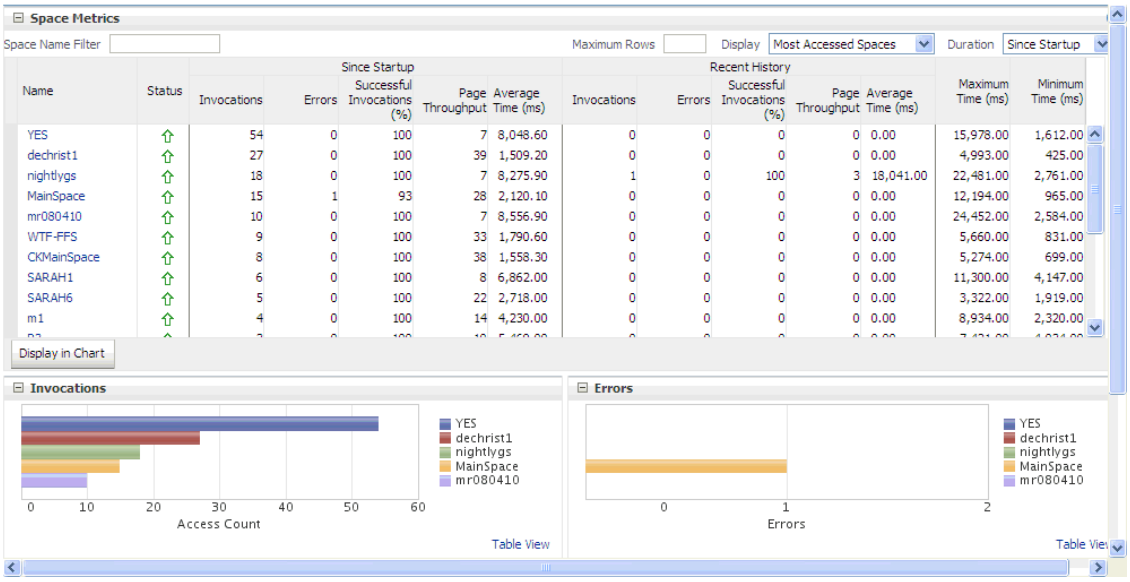
If you compare **Since Startup** metrics with **Recent History** metrics (last 15 minutes), you can determine whether performance has recently deteriorated, and if so, by how much.

27.1.11 Understanding Portal Metrics

(WebCenter Portal only) You can view live performance metrics for individual portals through Fusion Middleware Control, as shown in Figure 27–17. The metrics displayed on this page are described in Table 27–15 and Section 27.1.12.1, "Metrics Common to all Tools and Services."

> **Note:** Metrics for the Home portal are not included.

Figure 27–17 Portal Metrics



To monitor these metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

The table at the top of this page summarizes the status and performance of individual portals. Use the table to quickly see which portals are up and running, and to review their individual and relative performances.

Statistics become available when a portal is created and are updated every time a member accesses and uses the portal.

You can filter the data displayed in the following ways:

- **Space Name Filter** - Enter a full or partial search term, and then click the **Refresh** icon to refresh the list with all portals (previously referred to as *spaces*) for which a match is found in the display name. To display metrics for all portals, clear the search term and click **Refresh** again.
- **Maximum Rows** - Restrict the total number of portals displayed in the table.
- **Display** - Display metrics for the most popular portals, the slowest portals, or the portals experiencing the most errors. Depending on you selection, the table orders portals by:

  - Number of Invocations (most accessed portals)

  - Average Page Processing Time (slowest portals)

  - Number of Errors (portals with most errors)

- **Duration** - Display metric information collected since startup or in the last 15 minutes (Recent History).

The top five portals display in the chart.

Table 27–15 Portal Metrics

Metric	Description
Name	Names of portals that match your filter criteria (if any). If you do not specify filter criteria, all the portals are listed.

Table 27–15 (Cont.) Portal Metrics

Metric	Description
Status	Current status of each portal: ■ **Up** (Green Up Arrow) - Indicates that the last portal operation was successful. The portal is up and running. ■ **Down** (Red Down Arrow) - Indicates that the portal is not currently available or the last portal operation was unsuccessful due to an unexpected error or exception. User errors, such as an authentication failure, do not change the status to "Down". ■ **Unavailable** (Clock) - Status information is currently unavailable.
Invocations	Total number of portal invocations: - Since Startup - Recent History
Errors	Number of errors recorded.
Successful Invocations (%)	Percentage of portal invocations that succeeded: - Since Startup - Recent History If Successful Invocations (%) is below 100%, check the diagnostic logs to establish why portal requests are failing. See Section 28.2, "Viewing and Configuring Log Information."
Page Throughput	The average number of pages processed per minute for each portal: - Since Startup - Recent History
Average Time (ms)	The average time (in ms) to display pages in the portal: - Since Startup - Recent History
Maximum Time (ms)	Maximum time taken to display a page in the portal.
Minimum Time (ms)	Minimum time taken to display a page in the portal.

Space Metrics - Graphs

Use the graphs below the table to see information about portals (previously referred to as *spaces*):

- **Invocations** - Graph showing the most active/popular portals, that is, portals recording the most invocations.
- **Page Throughput** - Graph showing the average number of pages accessed per minute for each portal. Use this graph to identify portals with high (or low) page hit rates.
- **Average Processing Time** - Graph showing the average page response time (in milliseconds). Use this graph to identify portals with the best (or worst) page performance.
- **Errors** - Graph showing which portals are reporting the most errors. Use this graph to compare error rates.

To compare a different set of portals:

- Specify the appropriate filtering criteria.
- Select one or more portals in the table, and then click **Display in Chart**.

27.1.12 Understanding Tool and Service Metrics

This section includes the following topics:

- Section 27.1.12.1, "Metrics Common to all Tools and Services"
- Section 27.1.12.2, "Metrics Specific to a Particular Tool or Service"
- Section 27.1.12.3, "Troubleshooting Common Issues with Tools and Services"

27.1.12.1 Metrics Common to all Tools and Services

Fusion Middleware Control provides capabilities to monitor performance of tools and services used in WebCenter Portal and your Portal Framework applications in the following ways:

- Services summary: Summary of performance metrics for each tool or service used in WebCenter Portal or your Portal Framework application. Table 27–16 lists tools and services that use common performance metrics and Table 27–17 describes the common metrics.
- Most popular operations and response time for individual operations. Table 27–18 describes these metrics.
- Per operation metrics: Performance metrics for individual operations. Table 27–16 lists common performance metrics used to monitor performance of individual operations. Table 27–18 describes these metrics.

Table 27–16 Common Metrics for Tools and Services

Tool or Service	Services Summary (Since Startup and Recent History)	Per Operation Metrics (Since Startup and Recent History)
Announcements	The performance metrics include: ■ Status ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms)	The performance metrics include: ■ Most Popular Operations ■ Response Time ■ Invocations ■ Average Time (ms) ■ Maximum Time (ms) (Since Startup only)
BPEL Worklist	The performance metrics include: ■ Status ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms)	Not applicable

Table 27–16 (Cont.) Common Metrics for Tools and Services

Tool or Service	Services Summary (Since Startup and Recent History)	Per Operation Metrics (Since Startup and Recent History)
Discussion Forums	The performance metrics include: ■ Status ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms)	The performance metrics include: ■ Most Popular Operations ■ Response Time ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms) ■ Maximum Time (ms) (Since Startup only)
External Applications	The performance metrics include: ■ Status ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms)	The performance metrics include: ■ Most Popular Operations ■ Response Time ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms) ■ Maximum Time (ms) (Since Startup only)
Events	The performance metrics include: ■ Status ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms)	The performance metrics include: ■ Most Popular Operations ■ Response Time ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms) ■ Maximum Time (ms) (Since Startup only)
Import/Export	The performance metrics include: ■ Status ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms)	The performance metrics include: ■ Most Popular Operations ■ Response Time ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms) ■ Maximum Time (ms) (Since Startup only)

Table 27–16 (Cont.) Common Metrics for Tools and Services

Tool or Service	Services Summary (Since Startup and Recent History)	Per Operation Metrics (Since Startup and Recent History)
Instant Messaging and Presence (IMP)	The performance metrics include: ■ Status ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms)	The performance metrics include: ■ Most Popular Operations ■ Response Time ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms) ■ Maximum Time (ms) (Since Startup only)
Lists	The performance metrics include: ■ Status ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms)	The performance metrics include: ■ Most Popular Operations ■ Response Time ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms) ■ Maximum Time (ms) (Since Startup only)
Mail	The performance metrics include: ■ Status ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms)	The performance metrics include: ■ Most Popular Operations ■ Response Time ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms) ■ Maximum Time (ms) (Since Startup only)
Notes	The performance metrics include: ■ Status ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms)	The performance metrics include: ■ Most Popular Operations ■ Response Time ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms) ■ Maximum Time (ms) (Since Startup only)

Table 27–16 (Cont.) Common Metrics for Tools and Services

Tool or Service	Services Summary (Since Startup and Recent History)	Per Operation Metrics (Since Startup and Recent History)
Pages	The performance metrics include: ■ Status ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms)	The performance metrics include: ■ Most Popular Operations ■ Response Time ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms) ■ Maximum Time (ms) (Since Startup only)
People Connections	The performance metrics include: ■ Average Processing Time (ms) ■ Invocations ■ Successful Invocations (%)	The performance metrics include: ■ Most Popular Operations ■ Response Time ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms) ■ Maximum Time (ms) (Since Startup only)
Polls	The performance metrics include: ■ Average Processing Time (ms) ■ Invocations ■ Successful Invocations (%)	The performance metrics include: ■ Most Popular Operations ■ Response Time ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms) ■ Maximum Time (ms) (Since Startup only)
Recent Activity	The performance metrics include: ■ Average Time (ms) ■ Successful Invocations (%) ■ Invocations	Not available
RSS	The performance metrics include: ■ Status ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms)	Not available

Table 27–16 (Cont.) Common Metrics for Tools and Services

Tool or Service	Services Summary (Since Startup and Recent History)	Per Operation Metrics (Since Startup and Recent History)
Search	The performance metrics include: ■ Status ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms)	The performance metrics include: ■ Most Popular Operations ■ Response Time ■ Successful Invocations (%) ■ Invocations ■ Average Time (ms) ■ Maximum Time (ms) (Since Startup only)

Table 27–17 describes metrics used for monitoring performance of all operations.

Table 27–17 Description of Common Metrics - Summary (All Operations)

Metric	Description
Status	The current status of the tool or service: ■ **Up** (Green Up Arrow) - Indicates that a tool or service is up and running and the last operation was successful. ■ **Down** (Red Down Arrow) - Indicates that a tool or service is not currently available. The last operation was unsuccessful due to an unexpected error or exception. User errors, such as an authentication failure, do not change the status to Down. ■ **Unknown** (Clock) - Indicates that a tool or service cannot query the status of WebCenter Portal or your Portal Framework application for some reason. Maybe the managed server is down or the node cannot be reached due to a network issue. If a particular tool or service is "Down" or "Unknown", refer to Section 27.1.12.3, "Troubleshooting Common Issues with Tools and Services" for guidance on possible causes and actions.
Successful Invocations (%)	Percentage of service invocations that succeeded. Successful Invocations (%) equals the number of successful invocations divided by the invocation count: - Since Startup - Recent History If Successful Invocations (%) is below 100%, check the diagnostic logs to establish why service requests are failing. See Section 28.2, "Viewing and Configuring Log Information."
Invocations	Number of service invocations per minute: - Since Startup - Recent History This metric provides data on how frequently a particular tool or service is being invoked for processing of operations. Comparing this metric across services can help determine the most frequently used tools and services in the application.

Table 27–17 (Cont.) Description of Common Metrics - Summary (All Operations)

Metric	Description
Average Time (ms)	The average time taken to process operations associated with a tool or service. This metric can be used with the Invocations metric to assess the total time spent in processing operations. - Since Startup - Recent History Use this metric to determine the overall performance of tools and services. If this metric is out-of-bounds (the average time for operations is increasing or higher than expected), click individual names to view more detailed metric data.

Table 27–18 describes metrics used to monitor performance of each operation performed by a tool, service or component.

Table 27–18 Description of Common Metrics - Per Operation

Metric	Description
Most Popular Operations	The number of invocations per operation (displayed on a chart). The highest value on the chart indicates which operation is used the most. The lowest value indicates which operation is used the least.
Response Time	The average time to process operations associated with a service since WebCenter Portal or your Portal Framework application started up (displayed on a chart). The highest value on the chart indicates the worst performing operation. The lowest value indicates which operation is performing the best.
Operation	The operation being monitored. See Section 27.1.12.2, "Metrics Specific to a Particular Tool or Service."
Invocations	The number of invocations, per operation: - Since Startup - Recent History This metric provides data on how frequently a particular tool or service is being invoked for processing of operations. Comparing this metric across services can help determine the most frequently used service in the application.
Average Time (ms)	The average time taken to process each operation: - Since Startup* - Recent History *This information is also displayed on the **Response Time** chart.
Maximum Time (ms)	The maximum time taken to process each operation.

27.1.12.2 Metrics Specific to a Particular Tool or Service

This section describes *per operation* metrics for all tools, services and components. This section includes the following topics:

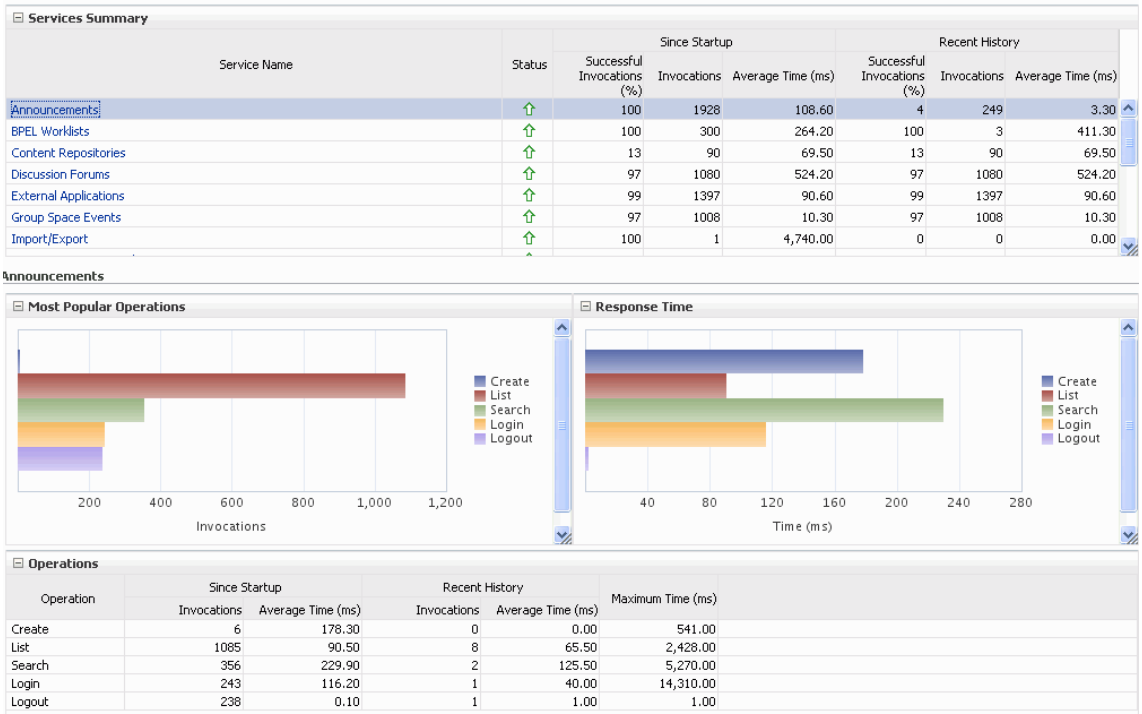
- Section 27.1.12.2.1, "Announcements Metrics"
- Section 27.1.12.2.2, "BPEL Worklist Metrics"

- Section 27.1.12.2.3, "Content Repository Metrics"
- Section 27.1.12.2.4, "Discussion Metrics"
- Section 27.1.12.2.6, "External Application Metrics"
- Section 27.1.12.2.5, "Events Metrics"
- Section 27.1.12.2.7, "Instant Messaging and Presence Metrics"
- Section 27.1.12.2.8, "Import and Export Metrics"
- Section 27.1.12.2.9, "List Metrics"
- Section 27.1.12.2.10, "Mail Metrics"
- Section 27.1.12.2.11, "Note Metrics"
- Section 27.1.12.2.12, "Page Operation Metrics"
- Section 27.1.12.2.13, "People Connection Metrics"
- Section 27.1.12.2.14, "Poll Metrics"
- Section 27.1.12.2.15, "RSS News Feed Metrics"
- Section 27.1.12.2.16, "Recent Activity Metrics"
- Section 27.1.12.2.17, "Search Metrics"

To access live performance metrics for WebCenter Portal or your Portal Framework application, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

27.1.12.2.1 Announcements Metrics Performance metrics associated with announcements (Figure 27–18) are described in Table 27–19 and Section 27.1.12.1, "Metrics Common to all Tools and Services."

Figure 27–18 Announcements Metrics



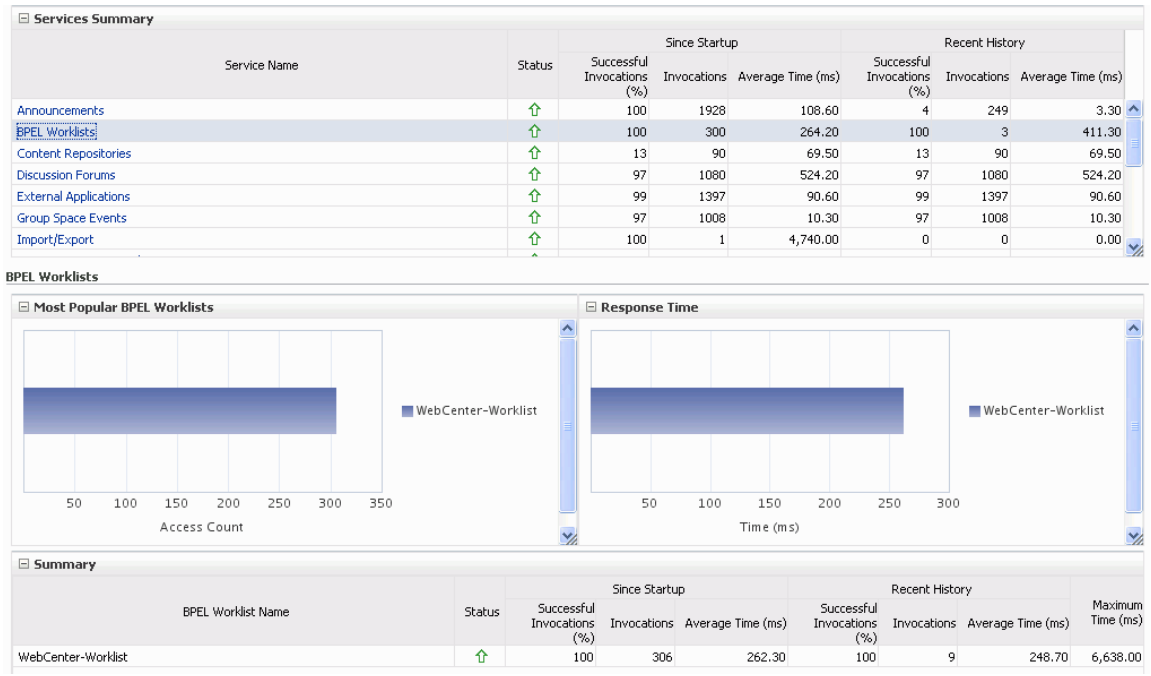
To monitor these metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

Table 27–19 Announcements - Operations Monitored

Operation	Description	Performance Issues - User Action
Login	Logs a WebCenter Portal user (accessing announcements) into the discussions server that is hosting announcements.	For specific causes, see Section 27.1.12.3.1, "Announcements - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Logout	Logs a WebCenter Portal user out of the discussions server that is hosting announcements.	For specific causes, see Section 27.1.12.3.1, "Announcements - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Search	Searches for terms within announcement text.	If announcement searches are failing, verify that announcement text contains the search terms. For other causes, see Section 27.1.12.3.1, "Announcements - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Create	Creates an announcement.	For specific causes, see Section 27.1.12.3.1, "Announcements - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
List	Retrieves a list of announcements.	For specific causes, see Section 27.1.12.3.1, "Announcements - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

27.1.12.2.2 BPEL Worklist Metrics Performance metrics associated with worklists (Figure 27–19) are described in Section 27.1.12.1, "Metrics Common to all Tools and Services."

Figure 27–19 BPEL Worklist Metrics



To monitor these metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

27.1.12.2.3 Content Repository Metrics Performance metrics associated with documents and Content Presenter (Figure 27–20 and Figure 27–21) are described in the following tables:

- Table 27–20, " Content Repository - Operations Monitored"
- Table 27–21, " Content Repository Metrics - Summary (All Repositories)"
- Table 27–22, " Content Repository Metrics - Operation Summary Per Repository"
- Table 27–23, " Content Repository Metrics - Operation Detail Per Repository"

Figure 27–20 Content Repository Metrics

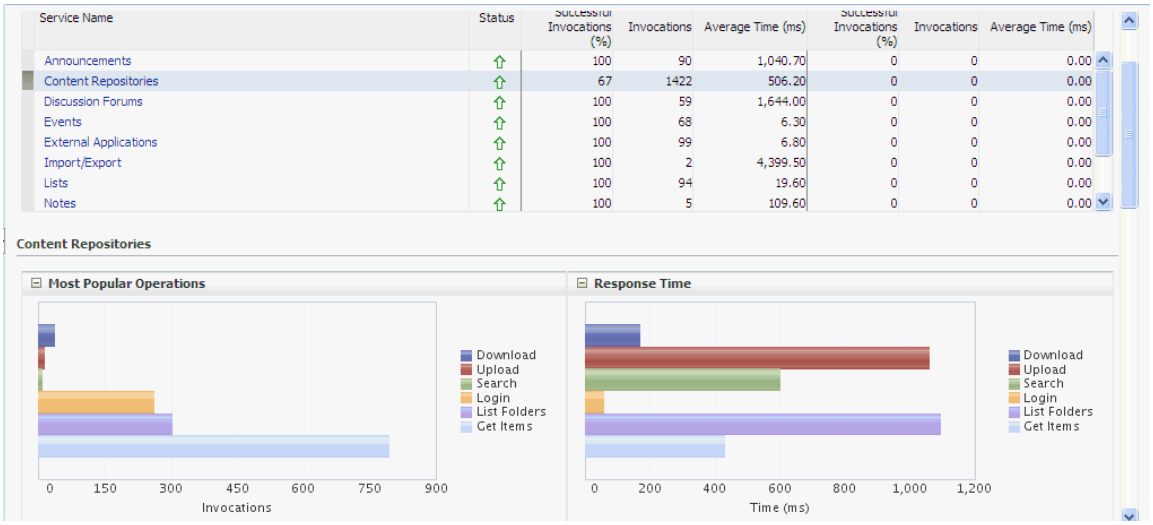
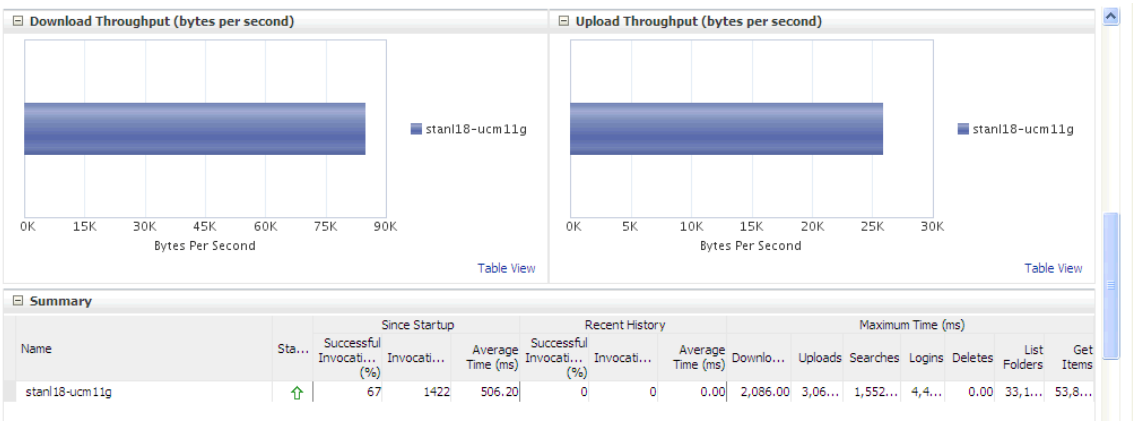


Figure 27–21 Content Repository Metrics - Per Operation



To monitor these metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

Table 27–20 Content Repository - Operations Monitored

Operation	Description	Performance Issues - User Action
Download	Downloads one or more documents from a content repository.	For specific causes, see Section 27.1.12.3.3, "Content Repository (Documents and Content Presenter) - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Upload	Uploads one or more documents to a content repository.	For specific causes, see Section 27.1.12.3.3, "Content Repository (Documents and Content Presenter) - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Search	Searches for documents stored in a content repository.	For specific causes, see Section 27.1.12.3.3, "Content Repository (Documents and Content Presenter) - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Login	Establishes a connection to the content repository and authenticates the user.	For specific causes, see Section 27.1.12.3.3, "Content Repository (Documents and Content Presenter) - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

Table 27–20 (Cont.) Content Repository - Operations Monitored

Operation	Description	Performance Issues - User Action
Delete	Deletes one or more documents stored in a content repository.	For specific causes, see Section 27.1.12.3.3, "Content Repository (Documents and Content Presenter) - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
List Folders	Lists folders stored in a content repository. This operation is specific to Content Presenter.	For specific causes, see Section 27.1.12.3.3, "Content Repository (Documents and Content Presenter) - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Get Items	Displays items, such as a document or image stored in a content repository. This operation is specific to Content Presenter.	For specific causes, see Section 27.1.12.3.3, "Content Repository (Documents and Content Presenter) - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

Table 27–21 Content Repository Metrics - Summary (All Repositories)

Metric	Description
Status	The current status of document tool: ▪ **Up** (Green Up Arrow) - Indicates that documents tool is up and running and the last operation was successful. ▪ **Down** (Red Down Arrow) - Indicates that documents tool is not currently available or service requests are failing. This also indicates that the last operation was unsuccessful due to an unexpected error or exception. User errors, such as an authentication failure, do not change the status to **Down**. If you are having problems with documents, check the diagnostic logs to establish why this tool is "Down". See Section 28.2, "Viewing and Configuring Log Information." Some typical causes of failure include: - Content repository is down or not responding. - Network connectivity issues exist between the application and one or more content repositories. - Connection configuration information associated with one or more content repositories is incorrect or no longer valid. ▪ **Unknown (Clock)** - Unable to query the status of the tool for some reason. Maybe the managed server is down or the node cannot be reached due to a network issues. To diagnose further, review the Admin Server log, and the managed server logs.

Table 27–21 (Cont.) Content Repository Metrics - Summary (All Repositories)

Metric	Description
Successful Invocations (%)	The percentage of document invocations that succeeded (Upload, Download, Search, Login, Delete): - Since Startup - Recent History If Successful Invocations (%) is below 100%, check the diagnostic logs to establish why service requests are failing. See Section 28.2, "Viewing and Configuring Log Information."
Invocations	The number of document invocations per minute (Upload, Download, Search, Login, Delete): - Since Startup - Recent History This metric provides data on how frequently a particular tool or service is being invoked for processing of operations. Comparing this metric across services can help determine the most frequently used tool or service in the application.
Average Time (ms)	The average time taken to process operations associated with documents (Upload, Download, Search, Login, Delete): - Since Startup - Recent History
Most Popular Operations	The number of invocations per operation (displayed on a chart). The highest value on the chart indicates which operation is used the most. The lowest value indicates which operations is used the least.
Response Time	The average time to process operations associated with documents since WebCenter Portal or your Portal Framework application started up (displayed on a chart). The highest value on the chart indicates the worst performing operation. The lowest value indicates which operations is performing the best.
Download Throughput (bytes per second)	The rate at which documents are downloaded.
Upload Throughput (bytes per second)	The rate at which documents are uploaded.

Table 27–22 Content Repository Metrics - Operation Summary Per Repository

Metric	Description
Status	The current status of the content repository: ■ **Up** (Green Up Arrow) - Indicates that the content repository is up and running and the last operation was successful. ■ **Down** (Red Down Arrow) - Indicates that the content repository is not currently available or service requests are failing. It also indicates that the last operation was unsuccessful due to an unexpected error or exception. User errors, such as an authentication failure, do not change the status to **Down**. If you are having problems with a content repository, check the diagnostic logs to establish why this service is "Down". See Section 28.2, "Viewing and Configuring Log Information." Some typical causes of failure include: - Content repository is down or not responding. - Network connectivity issues exist between the application and one or more content repositories. - Connection configuration information associated with one or more content repositories is incorrect or no longer valid. ■ **Unknown (Clock)** - Unable to query the status of the tool or service for some reason. Maybe the managed server is down or the node cannot be reached due to a network issues. To diagnose further, review the Admin Server log, and the managed server logs.
Successful Invocations (%)	The percentage of document invocations that succeeded (Upload, Download, Search, Login, Delete) for this content repository: - Since Startup - Recent History If Successful Invocations (%) is below 100%, check the diagnostic logs to establish why service requests are failing. See Section 28.2, "Viewing and Configuring Log Information."
Invocations	The number of document invocations per minute (Upload, Download, Search, Login, Delete) for this content repository: - Since Startup - Recent History This metric provides data on how frequently a particular tool or service is being invoked for processing of operations. Comparing this metric across tools and services can help determine the most frequently used tools and services in the application.
Average Time (ms)	The average time taken to process operations associated with documents (Upload, Download, Search, Login, Delete) for this content repository: - Since Startup - Recent History
Bytes Downloaded	The volume of data downloaded from this content repository.
Download Throughput (bytes per second)	The rate at which documents are downloaded from this content repository.
Bytes Uploaded	The volume of data uploaded to this content repository.

Table 27–22 (Cont.) Content Repository Metrics - Operation Summary Per Repository

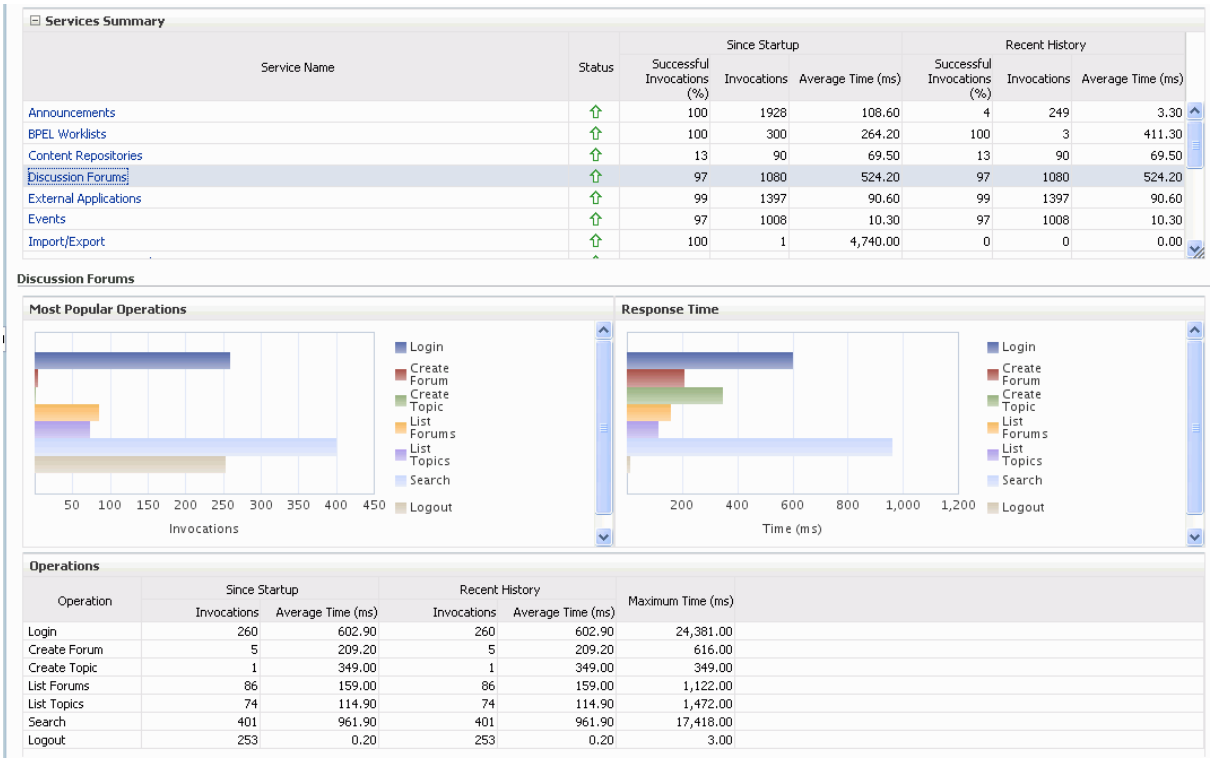
Metric	Description
Upload Throughput (bytes per second)	The rate at which documents are uploaded to this content repository.
Maximum Time (ms)	The maximum time to process operations associated with documents (Upload, Download, Search, Login, Delete) for this content repository.

Table 27–23 Content Repository Metrics - Operation Detail Per Repository

Metric	Description
Invocations	The number of invocations per document operation (Upload, Download, Search, Login, Delete): - Since Startup - Recent History This metric provides data on how frequently a particular service is being invoked for processing of operations. Comparing this metric across services can help determine the most frequently used services in the application.
Average Processing Time (ms)	The average time taken to process each operation associated with documents (Upload, Download, Search, Login, Delete): - Since Startup - Recent History

27.1.12.2.4 Discussion Metrics Performance metrics associated with discussions (Figure 27–22) are described in Table 27–24 and Section 27.1.12.1, "Metrics Common to all Tools and Services."

Figure 27–22 Discussion Metrics



To monitor these metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

Table 27–24 Discussions - Operations Monitored

Operation	Description	Performance Issues - User Action
Login	Logs a WebCenter Portal user (accessing discussions) into the discussions server that is hosting discussions forums.	For specific causes, see Section 27.1.12.3.4, "Discussions - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Logout	Logs a WebCenter Portal user out of the discussions server that is hosting discussion forums.	For specific causes, see Section 27.1.12.3.4, "Discussions - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

Table 27–24 (Cont.) Discussions - Operations Monitored

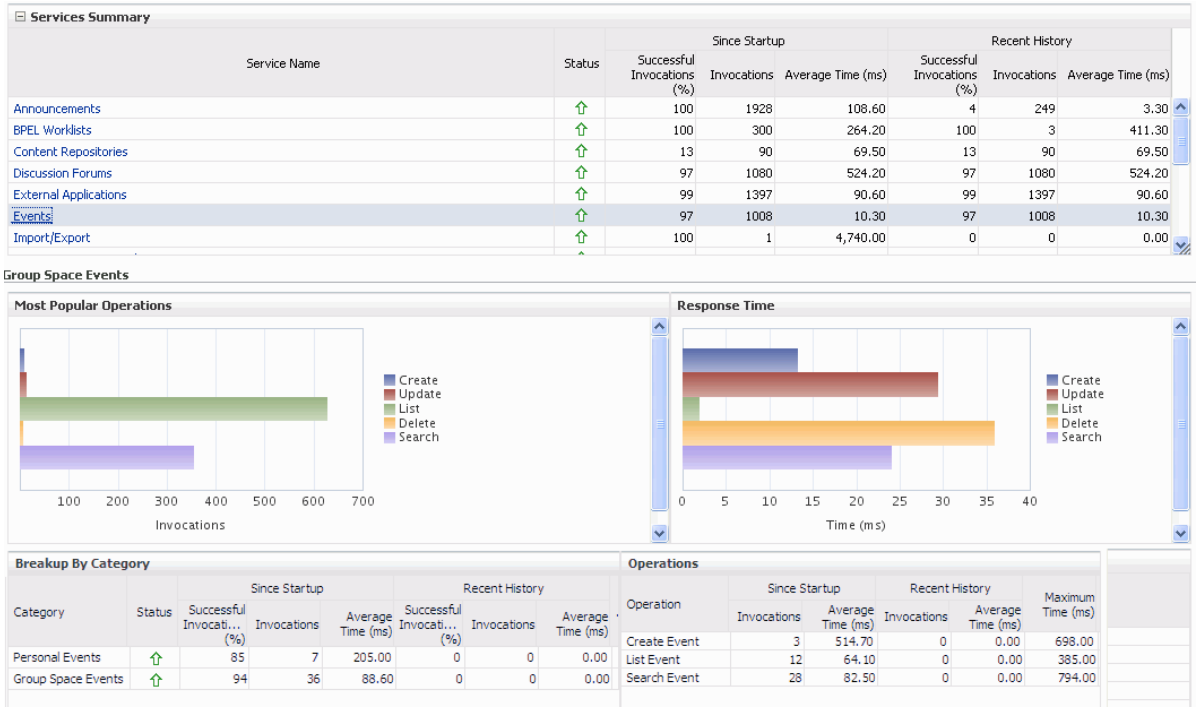
Operation	Description	Performance Issues - User Action
Create Forum	Creates a discussion forum in the discussions server, under a specific category.	If you are having problems creating forums, it may be due to: ■ Category under which discussion forums must be created has been deleted. ■ User does not have permissions to create discussion forums. For other specific causes, see Section 27.1.12.3.4, "Discussions - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Create Topic	Creates a topic in the discussions server, under a specific forum.	If you are having problems creating topics, it may be due to: ■ Discussion forum under which topics must be created has been deleted. ■ User does not have permissions to create topics. For other specific causes, see Section 27.1.12.3.4, "Discussions - Issues and Actions." For information on common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
List Forums	Retrieves a list of forums, under a specific category, from the discussion server.	If you are having problems viewing discussion forums, it may be due to: ■ User does not have permissions to view forums in the category. ■ Category from which to fetch forums has been deleted. For other specific causes, see Section 27.1.12.3.4, "Discussions - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

Table 27–24 (Cont.) Discussions - Operations Monitored

Operation	Description	Performance Issues - User Action
List Topics	Retrieves a list of topics, under a specific forum, from the discussion server.	If you are having problems viewing topics, it may be due to: ■ User does not have permissions to view topics in the forum. ■ Forum from which to fetch topics has been deleted. For other specific causes, see Section 27.1.12.3.4, "Discussions - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Search	Searches for terms within discussion forum text, in the discussions server.	If you are having problems searching forums, it may be due to: ■ No topic/messages exist with the specified search term. ■ Category or forum in which the search term object resides has been deleted. For other specific causes, see Section 27.1.12.3.4, "Discussions - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

27.1.12.2.5 Events Metrics Performance metrics associated with events are described in Table 27–25 and Section 27.1.12.1, "Metrics Common to all Tools and Services."

Figure 27–23 Events Metrics



To monitor these metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

Table 27–25 Events - Operations Monitored

Operation	Description	Performance Issues - User Action
Create Event	Creates a portal event or personal calendar event in the WebCenter Portal's repository.	For specific causes, see Section 27.1.12.3.6, "Events - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Update Event	Updates a portal event or personal calendar event stored in the WebCenter Portal's repository.	For specific causes, see Section 27.1.12.3.6, "Events - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Delete Event	Deletes a portal event or personal calendar event from the WebCenter Portal's repository.	For specific causes, see Section 27.1.12.3.6, "Events - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
List Event	Retrieves a list of events from the WebCenter Portal's repository.	For specific causes, see Section 27.1.12.3.6, "Events - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

Table 27–25 (Cont.) Events - Operations Monitored

Operation	Description	Performance Issues - User Action
Search Event	Searches for terms within event text.	For specific causes, see Section 27.1.12.3.6, "Events - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

27.1.12.2.6 External Application Metrics Performance metrics associated with external applications are described in Table 27–26 and Section 27.1.12.1, "Metrics Common to all Tools and Services."

Figure 27–24 External Application Metrics

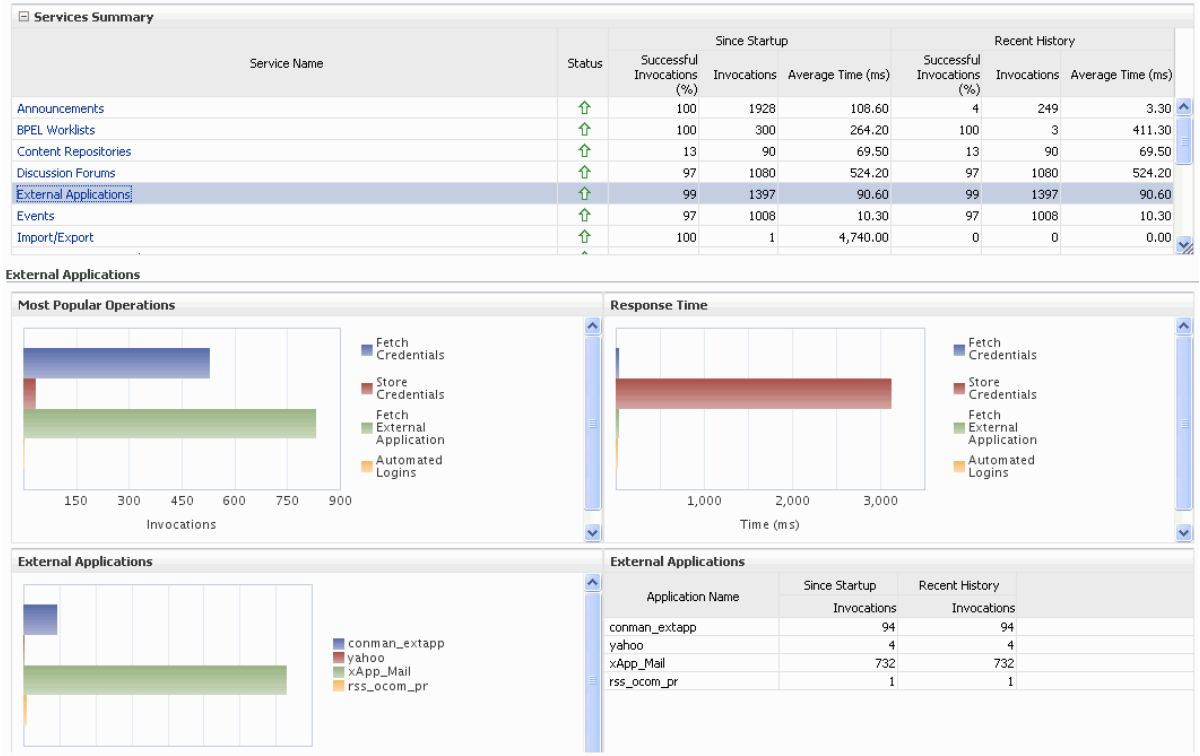
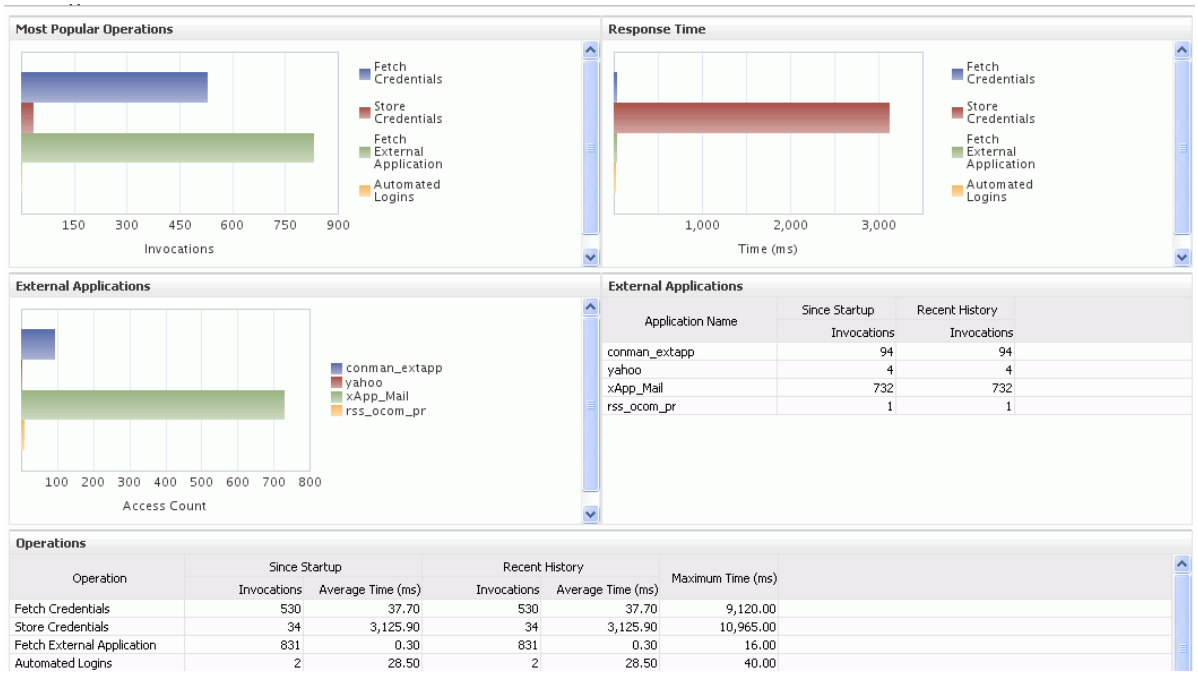


Figure 27–25 External Application Metrics - Per Operation



To monitor these metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

Table 27–26 External Applications - Operations Monitored

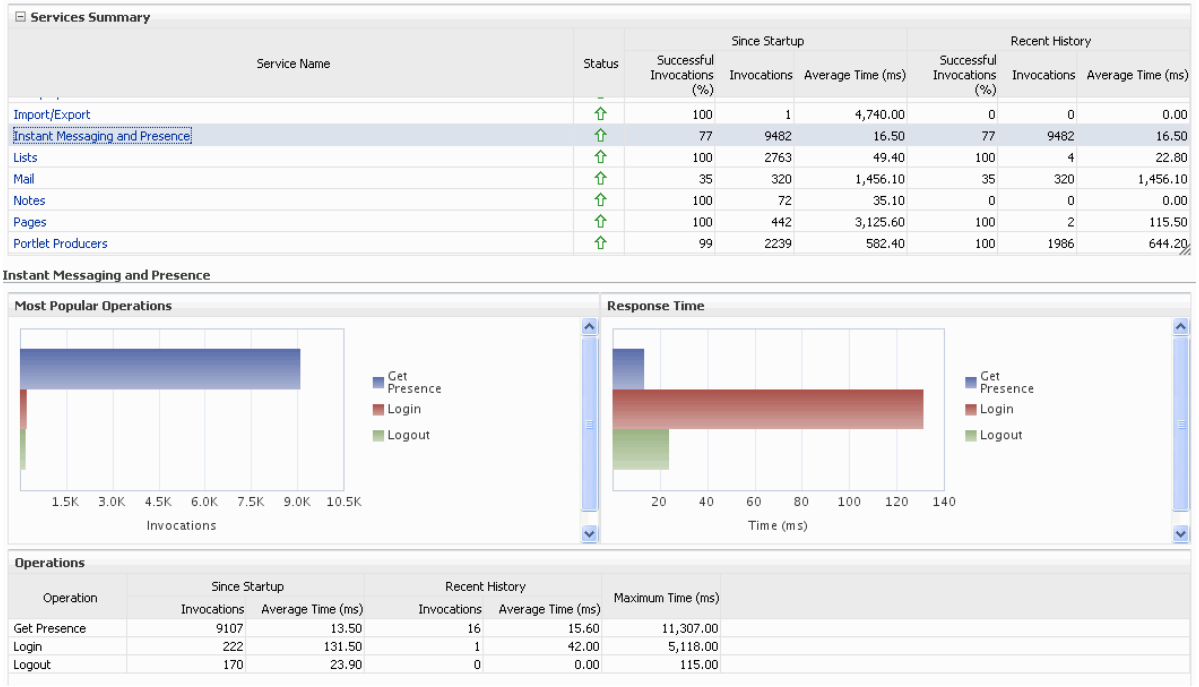
Operation	Description	Performance Issues - User Action
Fetch Credentials	Retrieves credentials for an external application.	For specific causes, see Section 27.1.12.3.5, "External Applications - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Store Credentials	Stores user credentials for an external application.	For specific causes, see Section 27.1.12.3.5, "External Applications - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Fetch External Application	Retrieves an external application.	For specific causes, see Section 27.1.12.3.5, "External Applications - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

Table 27–26 (Cont.) External Applications - Operations Monitored

Operation	Description	Performance Issues - User Action
Automated Logins	Logs a WebCenter Portal user in to an external application (using the automated login feature).	For specific causes, see Section 27.1.12.3.5, "External Applications - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

27.1.12.2.7 Instant Messaging and Presence Metrics Performance metrics associated with instant messaging and presence (Figure 27–26) are described in Table 27–27 and Section 27.1.12.1, "Metrics Common to all Tools and Services."

Figure 27–26 Instant Messaging and Presence Metrics



To monitor these metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

Table 27–27 Instant Messaging and Presence - Operations Monitored

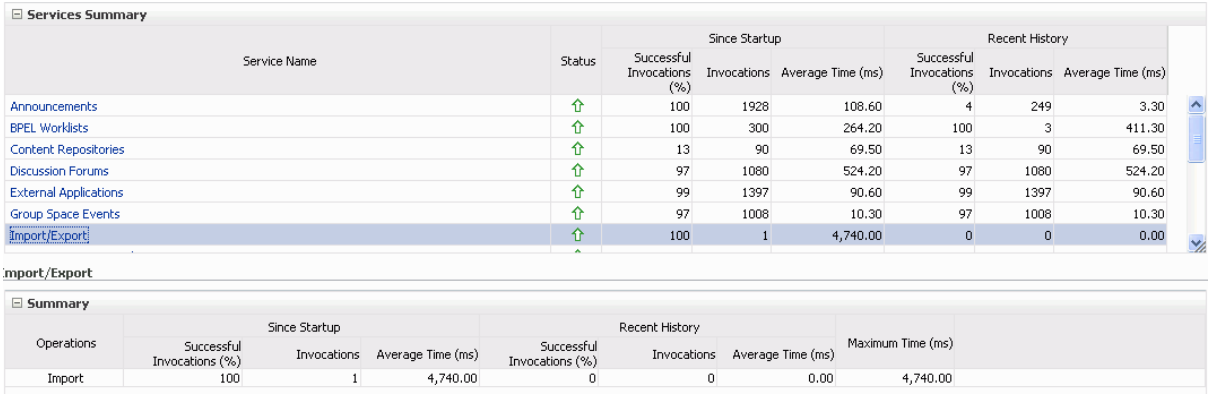
Operation	Description	Performance Issues - User Action
Get Presence	Retrieves user presence information from the instant messaging and presence server.	For specific causes, see Section 27.1.12.3.7, "Instant Messaging and Presence - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

Table 27–27 (Cont.) Instant Messaging and Presence - Operations Monitored

Operation	Description	Performance Issues - User Action
Login	Logs a WebCenter Portal user (accessing the instant messaging and presence) into the instant messaging and presence server.	For specific causes, see Section 27.1.12.3.7, "Instant Messaging and Presence - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Logout	Logs a WebCenter Portal user (accessing instant messaging and presence) out of the instant messaging and presence server.	For specific causes, see Section 27.1.12.3.7, "Instant Messaging and Presence - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

27.1.12.2.8 Import and Export Metrics Performance metrics associated with import and export (Figure 27–27) are described in Table 27–28 and Section 27.1.12.1, "Metrics Common to all Tools and Services." These metrics apply to WebCenter Portal only.

Figure 27–27 Import/Export Metrics



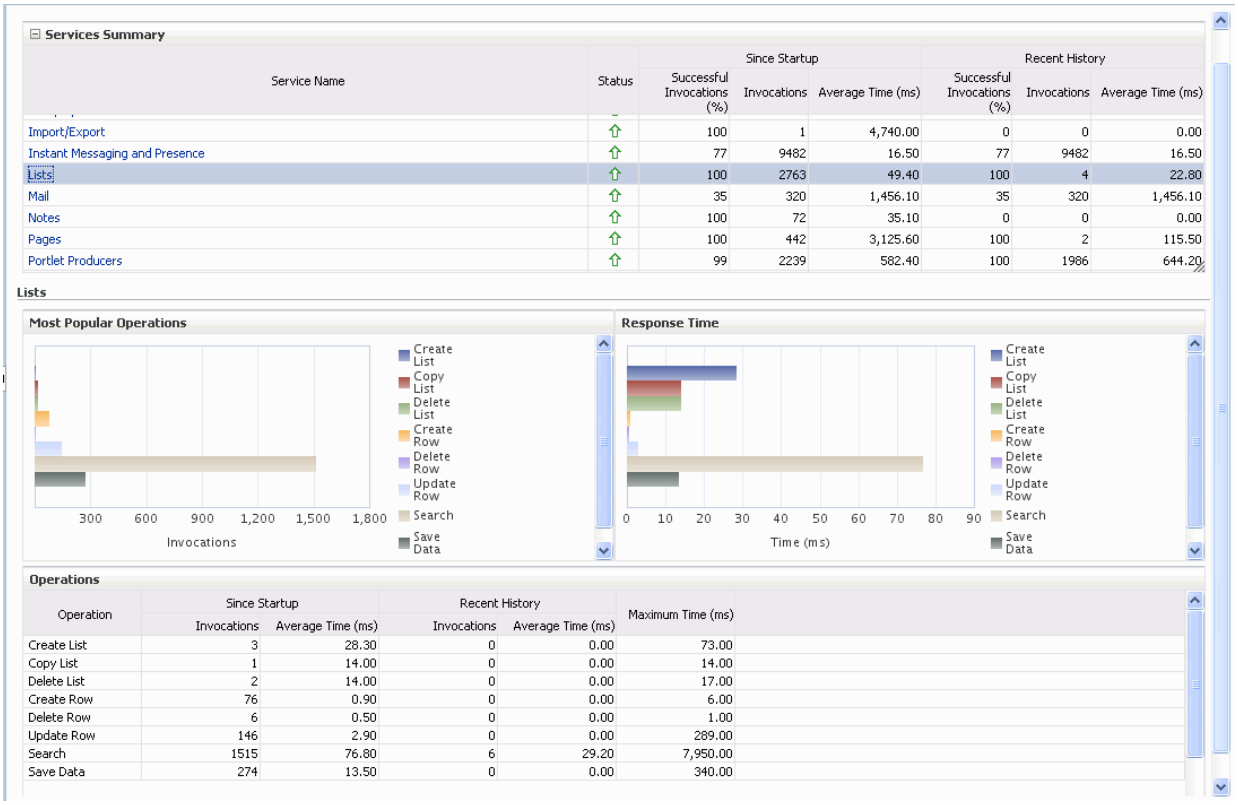
To monitor these metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

Table 27–28 Import/Export - Operations Monitored

Operation	Description	Performance Issues - User Action
Export	Exports an entireWebCenter Portal application.	For specific causes, see Section 27.1.12.3.8, "Import and Export - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Import	Imports an entire WebCenter Portal application.	For specific causes, see Section 27.1.12.3.8, "Import and Export - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

27.1.12.2.9 List Metrics (WebCenter Portal only) Performance metrics associated with lists (Figure 27–28) are described in Table 27–29 and Section 27.1.12.1, "Metrics Common to all Tools and Services."

Figure 27–28 List Metrics



To monitor these metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

Table 27–29 Lists- Operations Monitored

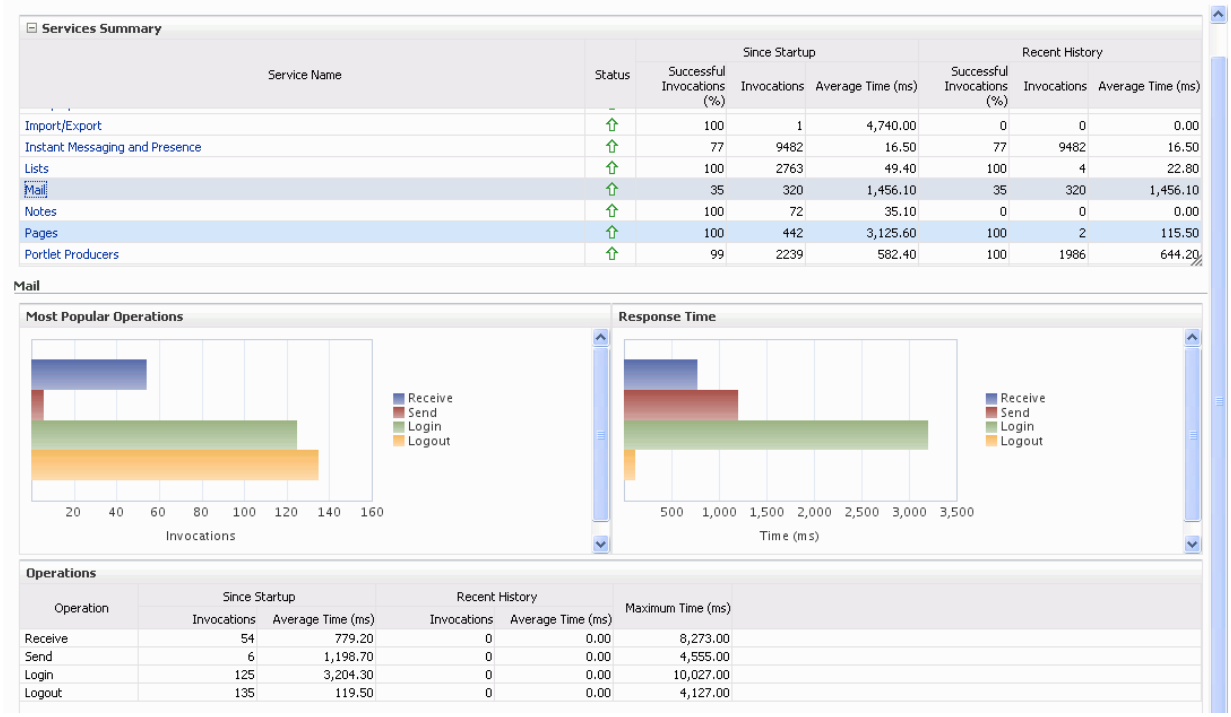
Operation	Description	Performance Issues - User Action
Create List	Creates a list in the user session. The Save Data operation commits new lists to the MDS repository.	For specific causes, see Section 27.1.12.3.9, "Lists - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Copy List	Copies a list and its data in the user session. The Save Data operation commits copied lists and list data to the MDS repository and the WebCenter Portal's repository (the database where list data is stored).	For specific causes, see Section 27.1.12.3.9, "Lists - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

Table 27–29 (Cont.) Lists- Operations Monitored

Operation	Description	Performance Issues - User Action
Delete List	Deletes a list and its data in the user session. The Save Data operation commits list changes to the MDS repository and the WebCenter Portal's repository (the database where list data is stored).	For specific causes, see Section 27.1.12.3.9, "Lists - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Create Row	Creates row of list data in the user session. The Save Data operation commits list data changes to the WebCenter Portal's repository (the database where list data is stored).	For specific causes, see Section 27.1.12.3.9, "Lists - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Update Row	Updates row of list data in the user session. The Save Data operation commits list data changes to the WebCenter Portal's repository (the database where list data is stored).	For specific causes, see Section 27.1.12.3.9, "Lists - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Delete Row	Deletes row of list data in the user session. The Save Data operation commits list data changes to the WebCenter Portal's repository (the database where list data is stored).	For specific causes, see Section 27.1.12.3.9, "Lists - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Search	Retrieves a list by its ID from the Metadata repository.	For specific causes, see Section 27.1.12.3.9, "Lists - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Save Data	Saves all changes to lists and list data (in the user session) to the Metadata Services repository and the WebCenter Portal's repository (the database where list information is stored).	For specific causes, see Section 27.1.12.3.9, "Lists - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

27.1.12.2.10 Mail Metrics Performance metrics associated with mail (Figure 27–29) are described in Table 27–30 and Section 27.1.12.1, "Metrics Common to all Tools and Services."

Figure 27–29 Mail Metrics



To monitor these metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

Table 27–30 Mail - Operations Monitored

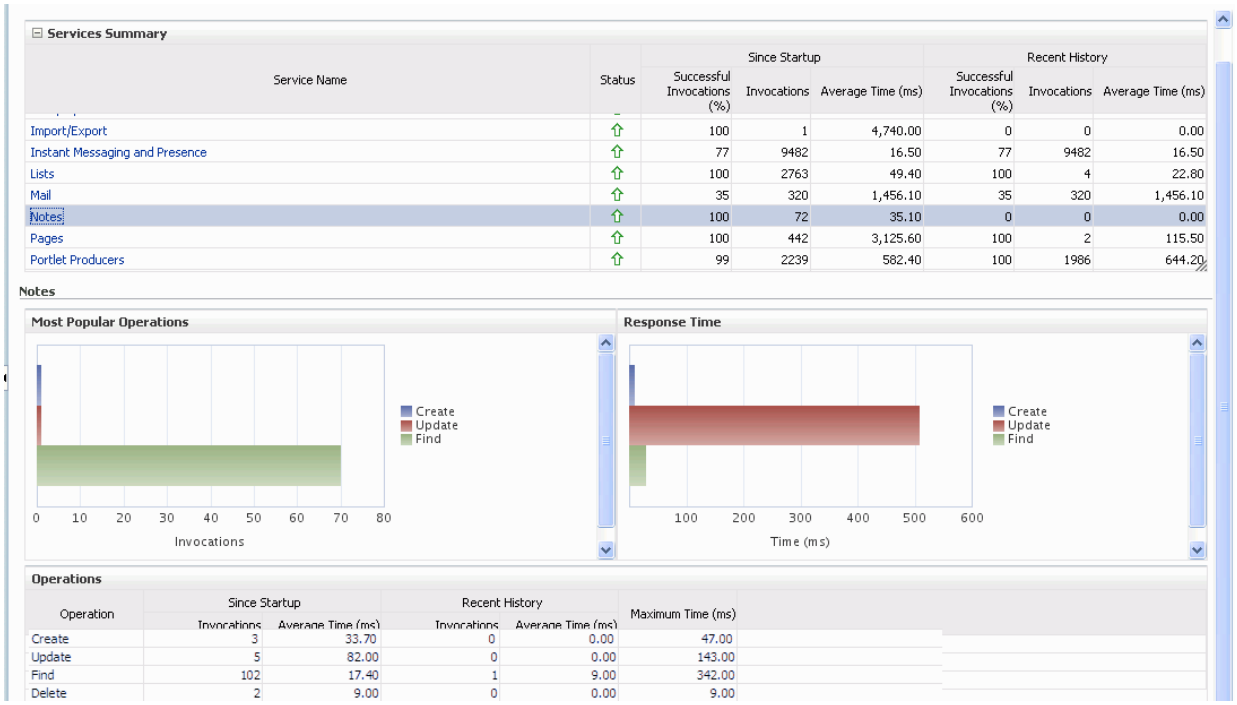
Operation	Description	Performance Issues - User Action
Login	Logs a WebCenter Portal user into the mail server that is hosting mail services.	For specific causes, see Section 27.1.12.3.10, "Mail - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Logout	Logs a WebCenter Portal user out of the mail server that is hosting mail services.	For specific causes, see Section 27.1.12.3.10, "Mail - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Receive	Receives a mail.	For specific causes, see Section 27.1.12.3.10, "Mail - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Send	Sends a mail.	For specific causes, see Section 27.1.12.3.10, "Mail - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

Table 27–30 (Cont.) Mail - Operations Monitored

Operation	Description	Performance Issues - User Action
Search	Searches for mail that contains a specific term.	For specific causes, see Section 27.1.12.3.10, "Mail - Issues and Actions." For information on common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

27.1.12.2.11 Note Metrics Performance metrics associated with notes (Figure 27–30) are described in Table 27–31 and Section 27.1.12.1, "Metrics Common to all Tools and Services."

Figure 27–30 Notes Metrics



To monitor these metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

Table 27–31 Notes - Operations Monitored

Operation	Description	Performance Issues - User Action
Create	Creates a personal note. The Save Changes operation commits new notes to the MDS repository.	For specific causes, see Section 27.1.12.3.11, "Notes - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

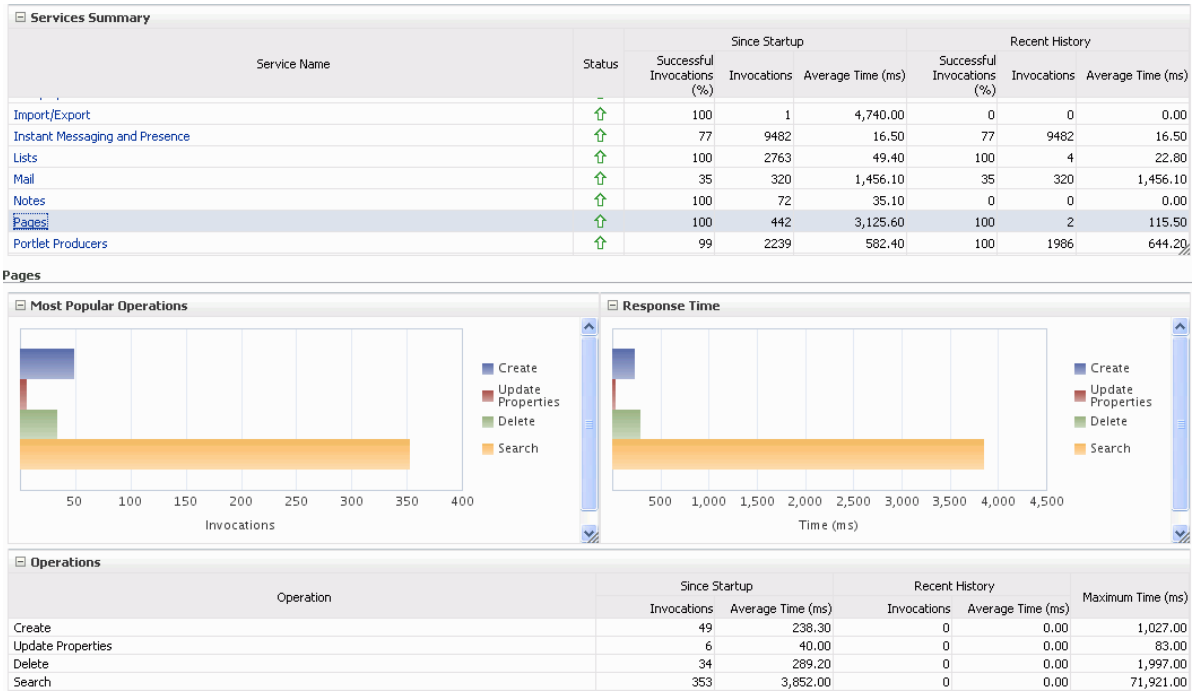
Table 27–31 (Cont.) Notes - Operations Monitored

Operation	Description	Performance Issues - User Action
Update	Updates a personal note. The Save Changes operation commits note updates to the MDS repository.	For specific causes, see Section 27.1.12.3.11, "Notes - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Find	Retrieves a note from the MDS repository.	For specific causes, see Section 27.1.12.3.11, "Notes - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Delete	Deletes a note from the MDS repository.	For specific causes, see Section 27.1.12.3.11, "Notes - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

27.1.12.2.12 Page Operation Metrics Performance metrics associated with the page operations (Figure 27–31) are described in Table 27–32 and Section 27.1.12.1, "Metrics Common to all Tools and Services."

Note: The *page operation* metrics discussed in this section are different from the *page request* metrics discussed in Section 27.1.5, "Understanding Page Request Metrics." Page operation metrics monitor page related operations such as creating pages. Whereas the page request metrics monitor individual page view/display requests (do not include page edit operations).

Figure 27–31 Page Operation Metrics



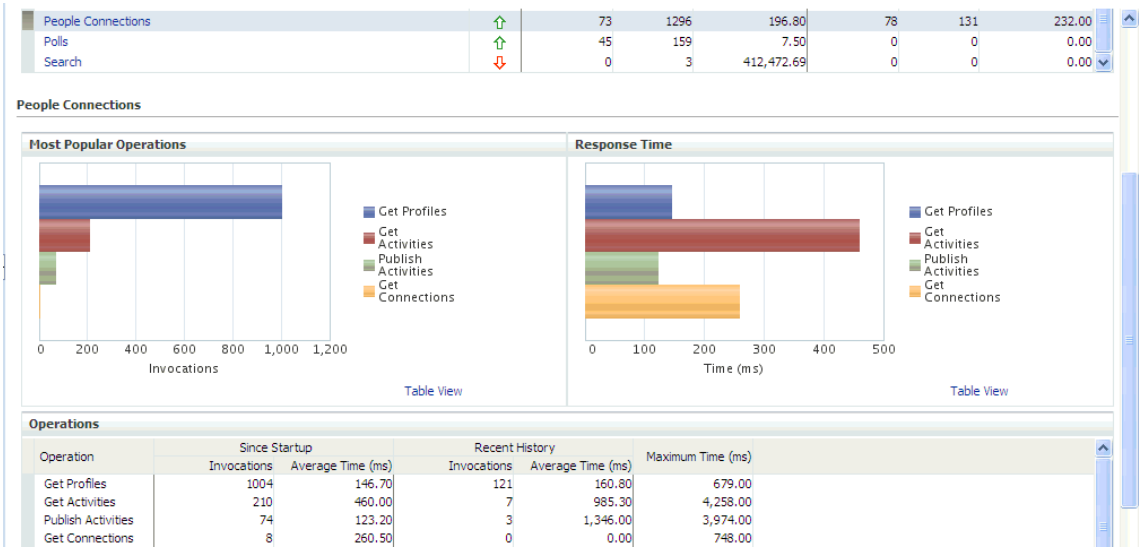
To monitor these metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

Table 27–32 Page Service - Operations Monitored

Operation	Description	Performance Issues - User Action
Create	Creates a page in WebCenter Portal or your Portal Framework application.	For specific causes, see Section 27.1.12.3.12, "Page Services - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Copy	Copies a page.	For specific causes, see Section 27.1.12.3.12, "Page Services - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Delete	Deletes a page.	For specific causes, see Section 27.1.12.3.12, "Page Services - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Search	Searches for pages that contain a specific term.	For specific causes, see Section 27.1.12.3.12, "Page Services - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

27.1.12.2.13 People Connection Metrics Performance metrics associated with people connections are described in Table 27–33 and Section 27.1.12.1, "Metrics Common to all Tools and Services."

Figure 27–32 People Connection Metrics



To monitor these metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

Table 27–33 People Connections - Operations Monitored

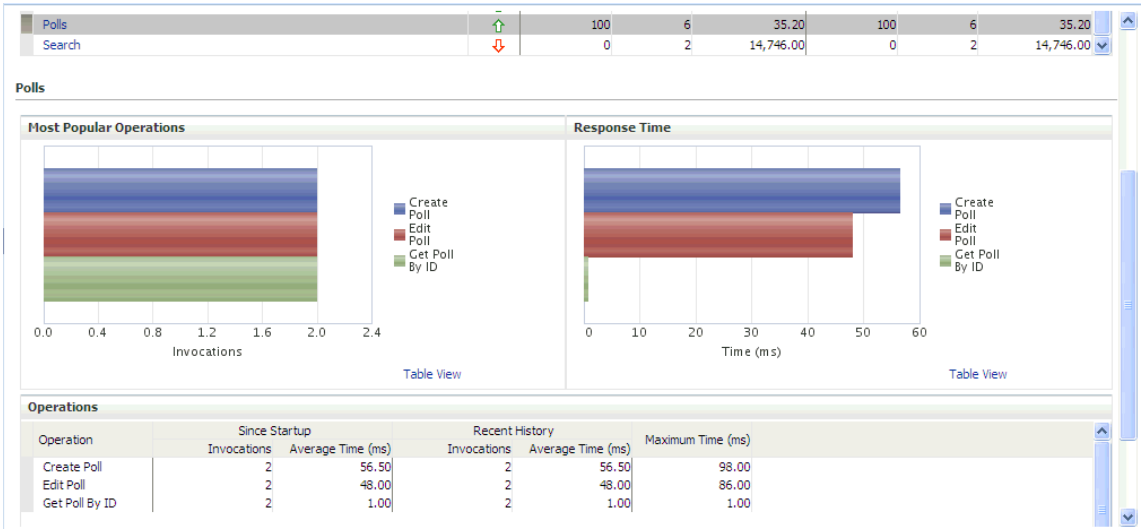
Operation	Description	Performance Issues - User Action
Get Profiles	Retrieves profiles of a user.	For specific causes, see Section 27.1.12.3.14, "People Connections - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Get Activities	Retrieves the activities based on the user filter options.	For specific causes, see Section 27.1.12.3.14, "People Connections - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Publish Activities	Publishes an activity in the user session and saves it in WebCenter Portal or your Portal Framework application.	For specific causes, see Section 27.1.12.3.14, "People Connections - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Get Messages	Retrieves the messages of the user.	For specific causes, see Section 27.1.12.3.14, "People Connections - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

Table 27–33 (Cont.) People Connections - Operations Monitored

Operation	Description	Performance Issues - User Action
Get Feedback	Retrieves the feedback of the user.	For specific causes, see Section 27.1.12.3.14, "People Connections - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Get Connections	Retrieves the connections of users.	For specific causes, see Section 27.1.12.3.14, "People Connections - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

27.1.12.2.14 Poll Metrics Performance metrics associated with polls (Figure 27–33) are described in Table 27–34 and Section 27.1.12.1, "Metrics Common to all Tools and Services."

Figure 27–33 Poll Metrics



To monitor these metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

Table 27–34 Polls - Operations Monitored

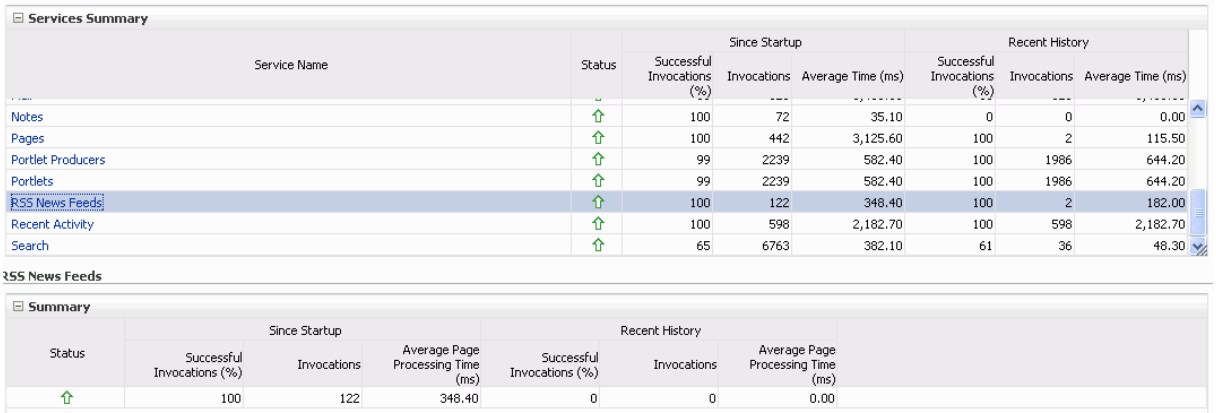
Operation	Description	Performance Issues - User Action
Create Poll	Creates a poll in WebCenter Portal or your Portal Framework application.	For specific causes, see Section 27.1.12.3.15, "Polls - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

Table 27–34 (Cont.) Polls - Operations Monitored

Operation	Description	Performance Issues - User Action
Edit Poll	Edit a poll in WebCenter Portal or your Portal Framework application.	For specific causes, see Section 27.1.12.3.15, "Polls - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Delete Poll	Deletes the ongoing poll.	For specific causes, see Section 27.1.12.3.15, "Polls - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Get Poll By ID	Displays the ongoing poll.	For specific causes, see Section 27.1.12.3.15, "Polls - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Submit Poll	Submits the ongoing poll.	For specific causes, see Section 27.1.12.3.15, "Polls - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."
Analyze Results	Analyzes the poll result.	For specific causes, see Section 27.1.12.3.15, "Polls - Issues and Actions." For common causes, see Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

27.1.12.2.15 RSS News Feed Metrics Performance metrics associated with RSS news feeds (Figure 27–34) are described in Section 27.1.12.1, "Metrics Common to all Tools and Services."

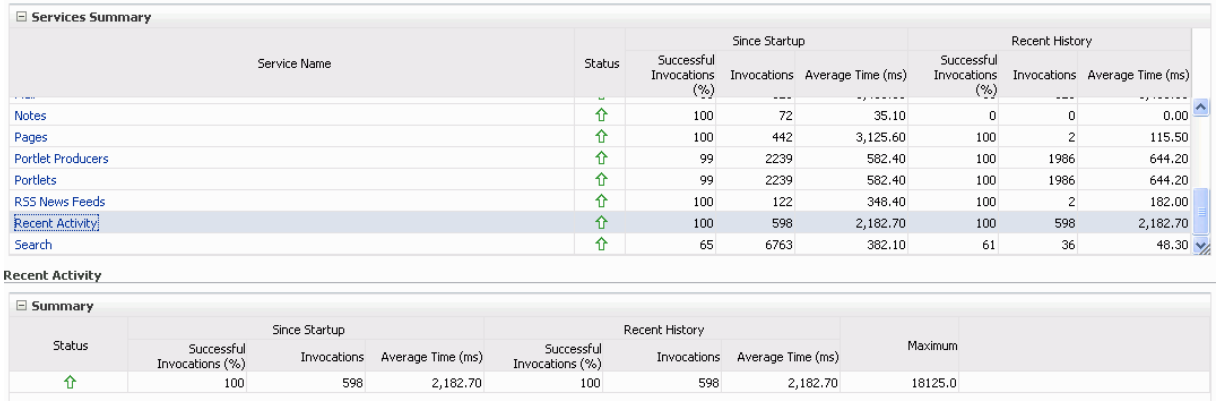
Figure 27–34 RSS News Feed Metrics



To monitor these metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

27.1.12.2.16 Recent Activity Metrics Performance metrics associated with recent activities (Figure 27–35) are described in Section 27.1.12.1, "Metrics Common to all Tools and Services."

Figure 27–35 Recent Activity Metrics



Services Summary

Service Name	Status	Since Startup			Recent History		
		Successful Invocations (%)	Invocations	Average Time (ms)	Successful Invocations (%)	Invocations	Average Time (ms)
Notes	⇧	100	72	35.10	0	0	0.00
Pages	⇧	100	442	3,125.60	100	2	115.50
Portlet Producers	⇧	99	2239	582.40	100	1986	644.20
Portlets	⇧	99	2239	582.40	100	1986	644.20
RSS News Feeds	⇧	100	122	348.40	100	2	182.00
Recent Activity	⇧	100	598	2,182.70	100	598	2,182.70
Search	⇧	65	6763	382.10	61	36	48.30

Recent Activity

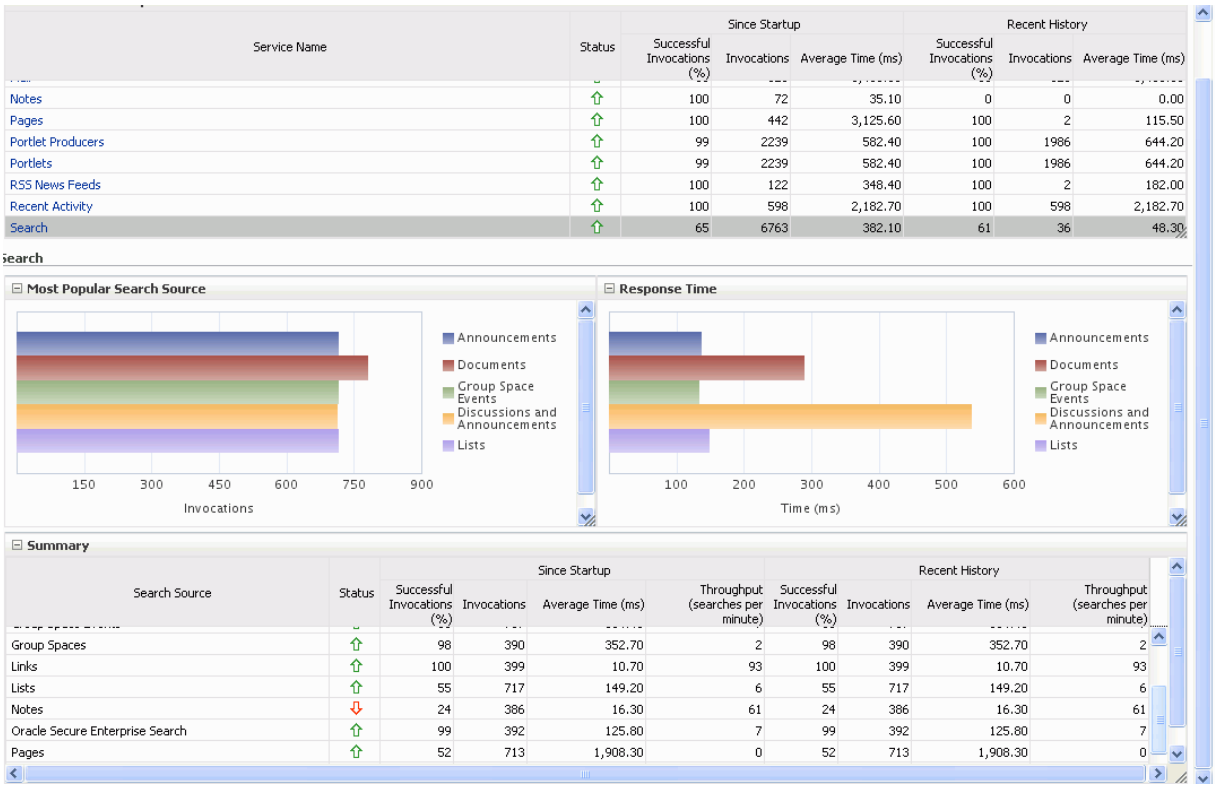
Summary

Status	Since Startup			Recent History			Maximum
	Successful Invocations (%)	Invocations	Average Time (ms)	Successful Invocations (%)	Invocations	Average Time (ms)	
⇧	100	598	2,182.70	100	598	2,182.70	18125.0

To monitor these metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

27.1.12.2.17 Search Metrics Performance metrics associated with search (Figure 27–36) are described in Table 27–35 and Section 27.1.12.1, "Metrics Common to all Tools and Services."

Figure 27–36 Search Metrics



To monitor these metrics through Fusion Middleware Control, see Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

Table 27–35 Search - Search Sources

Operation	Description
Announcements	Announcement text is searched.
Documents	Contents in files and folders are searched.
Discussion Forums	Forums and topics are searched.
WebCenter Portal	Contents saved in a portal, such as links, lists, notes, tags, and events are searched.
Space Events	Portal events (previously referred to as *space events*) are searched.
Links	Objects to which links have been created are searched (for example, announcements, discussion forum topics, documents, and events).
Lists	Information stored in lists is searched.
Notes	Notes text, such as reminders, is searched.
Oracle Secure Enterprise Search	Contents from the Document Library task flow, discussions, tag clouds, notes, and other tools and services are searched.
Pages	Contents added to application, personal, public, wiki, and blog pages are searched.

27.1.12.3 Troubleshooting Common Issues with Tools and Services

This section describes issues that you may have with individual tools and services and suggests actions you can take to address those issue.

See Also: Section 27.1.4, "Understanding Some Common Performance Issues and Actions"

This section includes the following topics:

- Section 27.1.12.3.1, "Announcements - Issues and Actions"
- Section 27.1.12.3.2, "BPEL Worklists - Issues and Actions"
- Section 27.1.12.3.3, "Content Repository (Documents and Content Presenter) - Issues and Actions"
- Section 27.1.12.3.4, "Discussions - Issues and Actions"
- Section 27.1.12.3.5, "External Applications - Issues and Actions"
- Section 27.1.12.3.6, "Events - Issues and Actions"
- Section 27.1.12.3.7, "Instant Messaging and Presence - Issues and Actions"
- Section 27.1.12.3.8, "Import and Export - Issues and Actions"
- Section 27.1.12.3.9, "Lists - Issues and Actions"
- Section 27.1.12.3.10, "Mail - Issues and Actions"
- Section 27.1.12.3.11, "Notes - Issues and Actions"
- Section 27.1.12.3.12, "Page Services - Issues and Actions"
- Section 27.1.12.3.13, "Portlets and Producers - Issues and Actions"
- Section 27.1.12.3.14, "People Connections - Issues and Actions"
- Section 27.1.12.3.15, "Polls - Issues and Actions"
- Section 27.1.12.3.16, "RSS News Feeds - Issues and Actions"
- Section 27.1.12.3.17, "Recent Activities - Issues and Actions"
- Section 27.1.12.3.18, "Search - Issues and Actions"

27.1.12.3.1 Announcements - Issues and Actions

If you are experiencing problems with announcements and the status is **Down**, check the diagnostic logs to establish why this service is unavailable. Some typical causes of failure include:

- Discussions server is down or not responding.
- Network connectivity issues exist between the application and the Discussions server.
- Connection configuration information associated with announcements is incorrect or no longer valid.

27.1.12.3.2 BPEL Worklists - Issues and Actions

If you are experiencing problems with worklists and the status is **Down**, check the diagnostic logs to establish why this service is unavailable. Some typical causes of failure include:

- BPEL server being queried is not available.
- Network connectivity issues exist between the application and the BPEL server.
- Connection configuration information associated with worklists is incorrect or no longer valid.

27.1.12.3.3 Content Repository (Documents and Content Presenter) - Issues and Actions

If you are experiencing problems with documents service and the status is **Down**, check the

diagnostic logs to establish why this service is unavailable. Also, do one of the following:

- For Content Server (Oracle WebCenter Content) and Oracle Portal, verify that the back-end server is up and running.
- For Content Server, verify that the socket connection is open for the client for which the service is not functioning properly. Check the list of IP addresses that are allowed to communicate with the Content Server through the Intradoc Server Port (IP Address Filter). For details, see the "Using Fusion Middleware Control to Modify Internet Configuration" section in *Oracle Fusion Middleware Administering Oracle WebCenter Content*.
- For Oracle Portal, verify the status of the JDBC connection using Oracle WebLogic Administration Console.
- (Functional check) Check logs on the back-end server. For Content Server, go to **Content Server** > **Administration** > **Log files** > **Content Server Logs.** For Oracle Portal use Fusion Middleware Control.
- (Functional check) Search for entries in the diagnostic log where the module name starts with `oracle.vcr`, `oracle.webcenter.content`, `oracle.webcenter.doclib`, and `oracle.stellent`. Specifically, the diagnostics log for the managed server on which WebCenter Portal is deployed located at:

  `DOMAIN_HOME/servers/`*`managed_server_name`*`/logs/<managed_server>-diagnostic.logs`

  For example, the diagnostics log for WebCenter Portal is named `WC_Spaces-diagnostic.log`. See Section 28.2, "Viewing and Configuring Log Information."

27.1.12.3.4 Discussions - Issues and Actions If you are experiencing problems with discussions and the status is **Down**, check the diagnostic logs to establish why this service is unavailable. Some typical causes of failure include:

- Discussions server is down or not responding.
- Network connectivity issues exist between the application and the discussion server.
- Connection configuration information associated with discussions is incorrect or no longer valid.

27.1.12.3.5 External Applications - Issues and Actions If you are experiencing problems with the External Applications service and the status is **Down**, check the diagnostic logs to establish why this service is unavailable. Some typical causes of failure include:

- Credential store is not configured for the application.
- Credential store that is configured, for example Oracle Internet Directory, is down or not responding.

27.1.12.3.6 Events - Issues and Actions If you are experiencing problems with events (portal events or personal events) and the status is **Down**, check the diagnostic logs to establish why this service is unavailable. Some typical causes of failure include:

- WebCenter Portal's repository is not available (the database where event information is stored).
- Network connectivity issues exist between the application and the WebCenter Portal's repository.

- Connection configuration information associated with events is incorrect or no longer valid.

27.1.12.3.7 Instant Messaging and Presence - Issues and Actions If you are experiencing problems with instant messaging and presence and the status is **Down**, check the diagnostic logs to establish why this service is unavailable. Some typical causes of failure include:

- Instant messaging and presence server is not available.
- Network connectivity issues exist between the application and the instant messaging and presence server.
- Connection configuration information associated with instant messaging and presence server is incorrect or no longer valid.

27.1.12.3.8 Import and Export - Issues and Actions If you are experiencing import and export problems and the status is **Down**, check the diagnostic logs to establish why this service is unavailable.

27.1.12.3.9 Lists - Issues and Actions If you are experiencing problems with lists and the status is **Down**, check the diagnostic logs to establish why this service is unavailable. Some typical causes of failure include:

- MDS repository or WebCenter Portal's repository, in which the data associated with lists is stored, is not available.
- Network connectivity issues exist between the application and the repository.

27.1.12.3.10 Mail - Issues and Actions If you are experiencing problems with mail and the status is **Down**, check the diagnostic logs to establish why this service is unavailable. Some typical causes of failure include:

- Mail server is not available.
- Network connectivity issues exist between the application and the mail server.
- Connection configuration information associated with mail server is incorrect or no longer valid.

27.1.12.3.11 Notes - Issues and Actions If you are experiencing problems with notes, check if the MDS repository is unavailable or responding slowly (the repository where note information is stored).

27.1.12.3.12 Page Services - Issues and Actions If you are experiencing problems with the page editing services and the status is **Down**, check the diagnostic logs to establish why this service is unavailable. Some typical causes of failure include:

- WebCenter Portal's repository is not available (the database where page information is stored).
- Network connectivity issues exist between the application and the WebCenter Portal's repository.

27.1.12.3.13 Portlets and Producers - Issues and Actions If you are experiencing problems with a portlet producer and the status is **Down**, check the diagnostic logs to establish why this service is unavailable. Some typical causes of failure include:

- Portlet producer server is down or not responding.

- Connection configuration information associated with the portlet producer is incorrect or no longer valid.
- Producer requests are timing out.
- There may be a problem with a particular producer, or the performance issue is due to a specific portlet(s) from that producer.

27.1.12.3.14 People Connections - Issues and Actions If you are experiencing problems with people connections and the status is **Down**, check the diagnostic logs to establish why this service is unavailable. Some typical causes of failure include:

- The service is down or not responding.
- WebCenter Portal's repository is not available (the database where people connection information is stored).
- Network connectivity issues exist between the application and the WebCenter Portal's repository.

27.1.12.3.15 Polls - Issues and Actions If you are experiencing problems with polls and the status is **Down**, check the diagnostic logs to establish why this service is unavailable. Some typical causes of failure include:

- The service is down or not responding.
- WebCenter Portal's repository is not available (the database where polls information is stored).
- Network connectivity issues exist between the application and the WebCenter Portal's repository.

27.1.12.3.16 RSS News Feeds - Issues and Actions If you are experiencing problems with RSS news feeds and the status is **Down**, check the diagnostic logs to establish why this service is unavailable. Some typical causes of failure include:

- RSS services are not available.
- A service being searched for activity data has failed, for example:
  - Unable to get discussions or announcement data - check the performance of discussions and announcements.
  - Unable to get list data - check the performance of lists.
  - Unable to get recent activities data - check the performance of recent activities.

27.1.12.3.17 Recent Activities - Issues and Actions If you are facing problems with recent activities and the status is **Down**, check the diagnostic logs to establish why this service is unavailable. Some typical causes of failure include:

- Recent activity services are not available.
- A service being searched for recent activity has failed.

27.1.12.3.18 Search - Issues and Actions If you are facing problems with search (a service executor) and the status is **Down**, check the diagnostic logs to establish why this executor is unavailable. Some typical causes of failure include:

- The repository of the executor is not available.
- Network connectivity issues exist between the application and the repository of the executor.

- Connection configuration information associated with the executor is incorrect or no longer valid.
- Content repositories being searched is currently unavailable.

27.2 Viewing Performance Metrics Using Fusion Middleware Control

Fusion Middleware Control monitors a wide range of performance metrics for WebCenter Portal or your Portal Framework application.

This section includes the following topics:

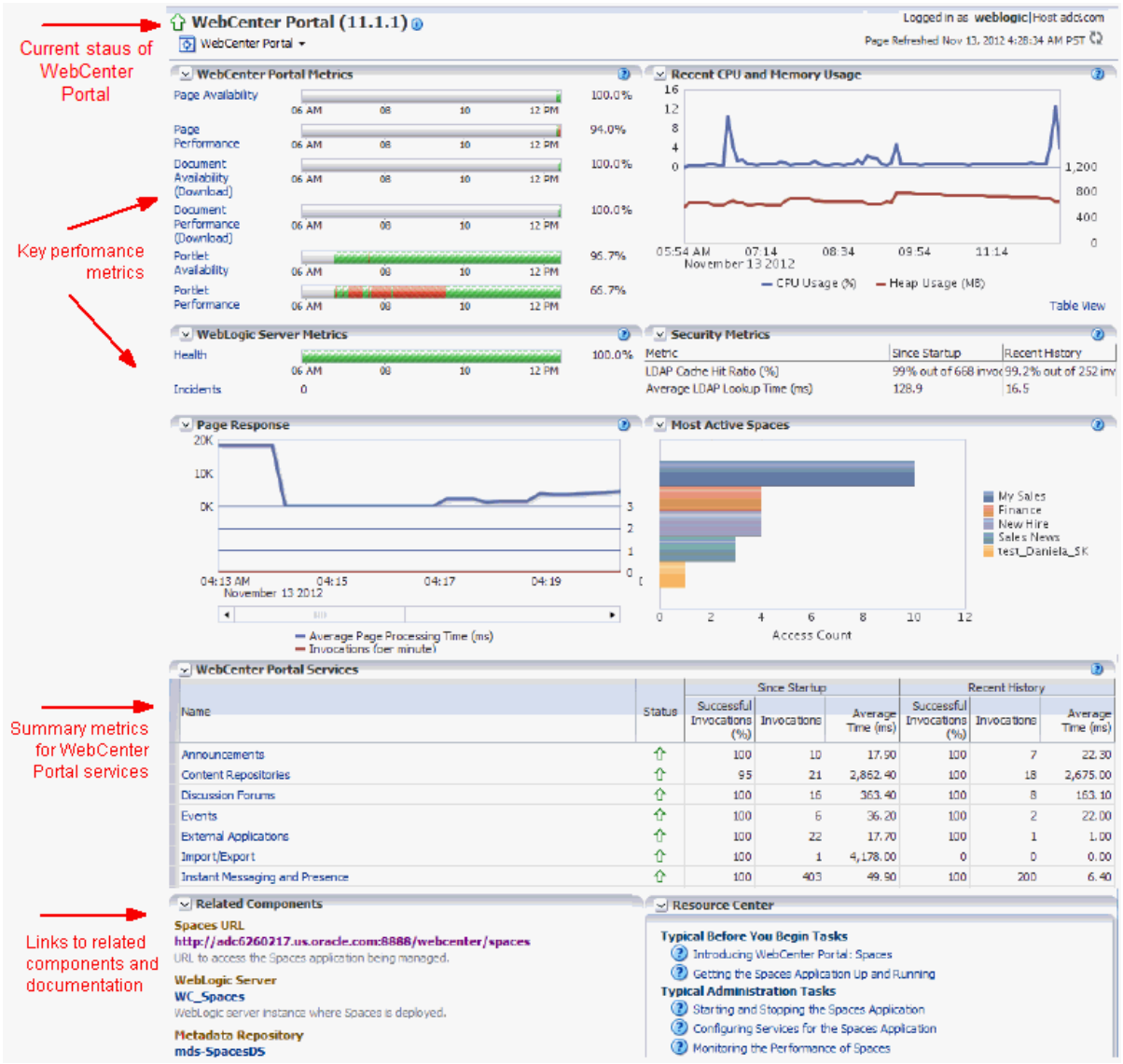
- Section 27.2.1, "Monitoring WebCenter Portal"
- Section 27.2.2, "Monitoring a Portal Framework Application"

27.2.1 Monitoring WebCenter Portal

Administrators can monitor the performance and availability of all the components and services that make up WebCenter Portal, and the application as a whole. These detailed metrics will help diagnose performance issues and, if monitored regularly, you will learn to recognize trends as they develop and prevent performance problems in the future.

Some key performance metrics display on the WebCenter Portal home page (Figure 27–37).

Figure 27–37 WebCenter Portal Home Page



The charts at the top of the page enable you to see at a glance whether the WebCenter Portal application is performing as expected or running slowly. You can drill down to more detailed metrics to troubleshoot problem areas and take corrective action. For guidance on what to look out for, see Section 27.1.3, "Using Key Performance Metric Data to Analyze and Diagnose System Health."

This section describes how to navigate around WebCenter Portal metric pages and includes the following topics:

- Section 27.2.1.1, "Monitoring Recent Performance Metrics for WebCenter Portal"
- Section 27.2.1.2, "Monitoring Portal Metrics"
- Section 27.2.1.4, "Monitoring Service Metrics for WebCenter Portal"
- Section 27.2.1.3, "Monitoring Page Metrics for WebCenter Portal"
- Section 27.2.1.5, "Monitoring All Metrics Through the Metrics Palette"

27.2.1.1 Monitoring Recent Performance Metrics for WebCenter Portal

To see how well WebCenter Portal or a particular portal is currently performing:

1. In Fusion Middleware Control Console, navigate to the home page for WebCenter Portal.

   See Section 6.2, "Navigating to the Home Page for WebCenter Portal."

2. Check the home page to see whether or not WebCenter Portal is operating as expected.

   For guidance on what to look out for, see Section 27.1.3, "Using Key Performance Metric Data to Analyze and Diagnose System Health."

3. Drill down to more detailed metrics by clicking links on the home page, such as Page Performance, Portlet Availability, Health, and so on.

   Alternatively, access detailed recent metrics through the following menu options:

   - **WebCenter Portal > Monitoring >Recent Page Metrics**
   - **WebCenter Portal > Monitoring >Recent Document Metrics**
   - **WebCenter Portal > Monitoring >Recent Portlet Metrics**
   - **WebCenter Portal > Monitoring >Recent WebLogic Server Metrics**

   For more information about the metrics on the these pages, see "Understanding Page Request Metrics", "Understanding Document Metrics", "Understanding Portlet Producer Metrics", and "Understanding WebLogic Server Metrics".

27.2.1.2 Monitoring Portal Metrics

To access performance metrics for portals created in WebCenter Portal:

1. In Fusion Middleware Control Console, navigate to the home page for WebCenter Portal:

   See Section 6.2, "Navigating to the Home Page for WebCenter Portal."

2. From the **WebCenter Portal** menu, select **Monitoring** > **Overall Space Metrics**.

   To learn more about the metrics displayed, see "Understanding Portal Metrics". See Section 27.1.4, "Understanding Some Common Performance Issues and Actions."

3. Drill down to detailed page metrics for a particular portal or compare a specific set of portals:

   - To see detailed performance information for a specific portal (previously referred to as *spaces*):

     In the **Space Name Filter** field, enter the name of a portal, then press **[Enter]**. For information about portal filtering options, see Section 27.1.11, "Understanding Portal Metrics."

     OR

     In the **Name** column, click the portal name (link) for which you want to display performance metrics.

     In both cases, page metrics for the selected portal display.

   - To compare the performance of one or more portals, select one or more rows in the table, and select **Display in Chart**.

27.2.1.3 Monitoring Page Metrics for WebCenter Portal

To access page metrics:

1. In Fusion Middleware Control Console, navigate to the home page for WebCenter Portal.

   See Section 6.2, "Navigating to the Home Page for WebCenter Portal."

2. Review page availability/performance charts on the home page to see whether page requests are currently responding as expected.

   To drill down to more detailed information, click **Page Availability**, **Page Performance**, or select **Monitoring** > **Recent Page Metrics**. For more information about the metrics displayed, see Section 27.1.5.2, "Recent Page Metrics."

3. To monitor page performance since start up, select **Monitoring** > **Overall Page Metrics**.

   You can view metrics for a particular page, all pages, or a specific set of pages. For more information about the metrics displayed and page filtering options, see Section 27.1.5.3, "Overall Page Metrics."

4. To monitor the performance of page editing operations, select **Monitoring** > **Overall Service Metrics** and then click **Pages** in the table.

   For information about the metrics displayed, see Section 27.1.12.2.12, "Page Operation Metrics."

27.2.1.4 Monitoring Service Metrics for WebCenter Portal

To access service metrics for the WebCenter Portal application:

1. In Fusion Middleware Control Console, navigate to the home page for WebCenter Portal.

   See Section 6.2, "Navigating to the Home Page for WebCenter Portal."

2. From the **WebCenter Portal** menu, select **Monitoring** > **Overall Service Metrics**.

   Use **Services Summary** at the top of the **WebCenter Portal Service Metrics** page to quickly see which services are up and running, and to review individual and relative performances of those services used by WebCenter Portal.

   Metrics become available when a tool, service, application, or portlet is accessed for the first time. If a service is not configured or has never been used it will not appear in the **Summary** table.

3. Click the name of a service to drill down to more detailed metrics.

   To learn more about individual metrics, see Section 27.1.12.2, "Metrics Specific to a Particular Tool or Service." See also, Section 27.1.12.3, "Troubleshooting Common Issues with Tools and Services."

27.2.1.5 Monitoring All Metrics Through the Metrics Palette

To access and chart any performance metric collected for WebCenter Portal:

1. In Fusion Middleware Control Console, navigate to the home page for WebCenter Portal.

   See Section 6.2, "Navigating to the Home Page for WebCenter Portal."

2. From the **WebCenter Portal** menu, select **Monitoring** > **Performance Summary**.

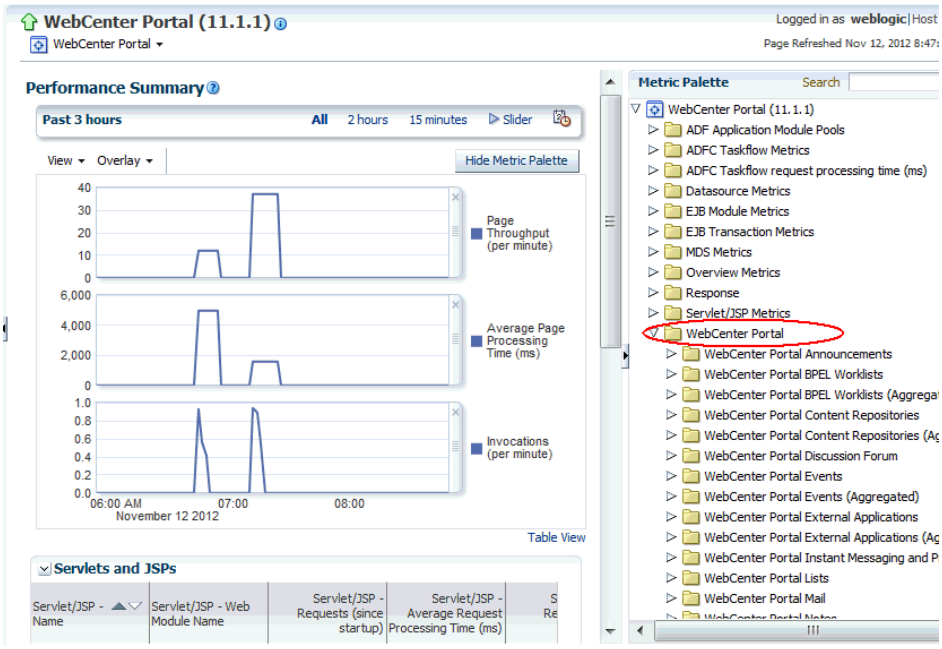
Use the **Show Metric Palette** button at the top of the **Performance Summary** page to display the **Metric Palette**. This palette enables you to select and monitor individual metrics.

3. In the **Metric Palette**, expand the folders under **WebCenter Portal** and then select the metric check boxes to monitor the metric in graphical or tabular format.

   Figure 27–38 shows the Performance Summary page and Metric Palette. In addition to **WebCenter Portal** performance metrics, the Metric Palette also displays general performance metrics associated with any J2EE application, for example, **ADF Application Module Pool** metrics.

   To display online help for any metric, right-click the required directory or any metric in the directory and select **Help**.

Figure 27–38 WebCenter Portal - Performance Summary and Metric Palette



27.2.2 Monitoring a Portal Framework Application

Administrators can monitor the performance and availability of all the components and services that make up Portal Framework applications built JDeveloper, and the application as a whole. These detailed metrics will help diagnose performance issues and, if monitored regularly, you will learn to recognize trends as they develop and prevent performance problems in the future.

This section includes the following topics:

- Section 27.2.2.1, "Monitoring Recent Performance Metrics for a Portal Framework Application"
- Section 27.2.2.2, "Monitoring Service Metrics for a Portal Framework Application"
- Section 27.2.2.3, "Monitoring Page Metrics for a Portal Framework Application"
- Section 27.2.2.4, "Monitoring All Metrics Through the Metrics Palette"

27.2.2.1 Monitoring Recent Performance Metrics for a Portal Framework Application

To see how well your Portal Framework application is currently performing:

1. In Fusion Middleware Control Console, navigate to the home page for the Portal Framework application.

   See Section 6.3, "Navigating to the Home Page for Portal Framework Applications."

2. Check the **Response and Load** chart to review the latest page performance and verify whether the number of people currently using the application is impacting performance. See Section 27.1.10, "Understanding Page Response and Load Metrics."

3. Drill down to more detailed metrics through the following menu options:

   - **Application Deployment> WebCenter Portal > Recent Page Metrics**
   - **Application Deployment> WebCenter Portal > Recent Document Metrics**
   - **Application Deployment> WebCenter Portal > Recent Portlet Metrics**
   - **Application Deployment> WebCenter Portal > Recent WebLogic Server Metrics**

   For more guidance on what to look out for, see Section 27.1.3, "Using Key Performance Metric Data to Analyze and Diagnose System Health."

   For detailed information about the metrics on the these pages, see "Understanding Page Request Metrics", "Understanding Document Metrics", "Understanding Portlet Producer Metrics", and "Understanding WebLogic Server Metrics".

27.2.2.2 Monitoring Service Metrics for a Portal Framework Application

To access performance metrics for any services that are used in a Portal Framework application:

1. In Fusion Middleware Control Console, navigate to the home page for Portal Framework applications.

   See Section 6.3, "Navigating to the Home Page for Portal Framework Applications."

2. From the **Application Deployment** menu, select **WebCenter Portal** > **Overall Service Metrics**.

   Use the **Services Summary** at the top of the **WebCenter Service Metrics** page to quickly see which services are up and running, and to review individual and relative performances of all the services used by the Portal Framework application.

   Statistics become available when a service, application, or portlet is accessed for the first time. If a service is not configured or has never been used it will not appear in the Services Summary table.

3. Click the name of a service to drill down to more detailed metrics.

   To learn more about individual metrics, see Section 27.1.12.2, "Metrics Specific to a Particular Tool or Service." See also, Section 27.1.12.3, "Troubleshooting Common Issues with Tools and Services."

27.2.2.3 Monitoring Page Metrics for a Portal Framework Application

To access performance metrics for pages displayed in a Portal Framework application:

1. In Fusion Middleware Control Console, navigate to the home page for Portal Framework applications.

   See Section 6.3, "Navigating to the Home Page for Portal Framework Applications."

2. To see whether page requests are currently responding as expected, select **Application Deployment** > **WebCenter Portal > Recent Page Metrics**.

   Review the recent page availability and performance charts. For more information about the metrics displayed, see Section 27.1.5.2, "Recent Page Metrics."

3. To monitor page performance since start up, select **Application Deployment > WebCenter Portal > Overall Page Metrics**.

   You can view metrics for a particular page, all pages, or a specific set of pages. For more information about the metrics displayed and page filtering options, see Section 27.1.5.3, "Overall Page Metrics."

4. To monitor the performance of the page editing operations, select **Application Deployment> WebCenter Portal** > **Overall Service Metrics** and then click **Pages** in the table.

   For information about the metrics displayed, see Section 27.1.12.2.12, "Page Operation Metrics."

27.2.2.4 Monitoring All Metrics Through the Metrics Palette

To access and chart any performance metric collected for a Portal Framework application:

1. In Fusion Middleware Control Console, navigate to the home page for Portal Framework applications.

   See Section 6.3, "Navigating to the Home Page for Portal Framework Applications."

2. From the **Application Deployment** menu, select **Performance Summary**.

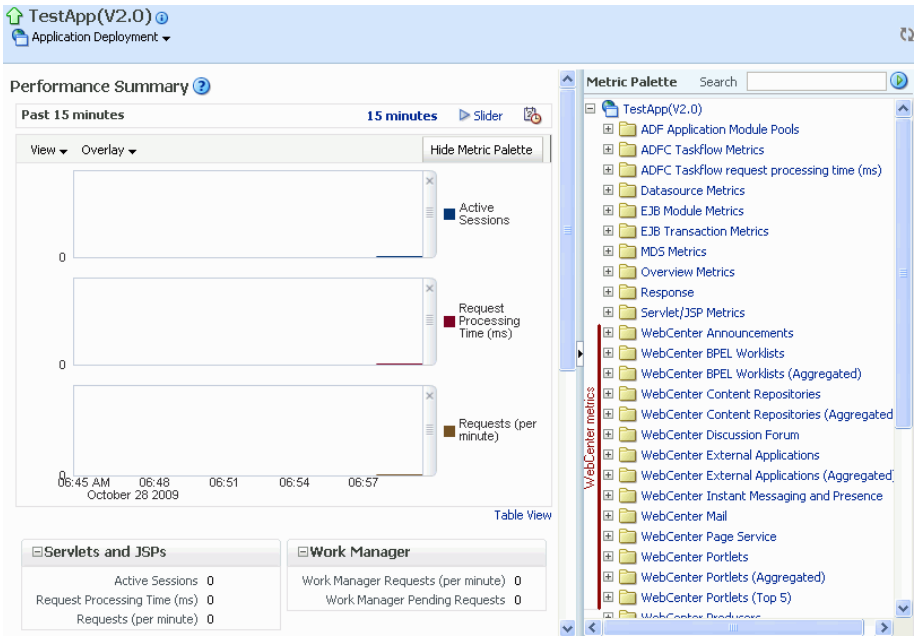
   Use the **Show Metric Palette** button at the top of the **Performance Summary** page to display the **Metric Palette**. This palette enables you to select and monitor individual metrics.

3. In the **Metric Palette**, expand the folders under **WebCenter Portal** and then select the metric check boxes to monitor the metric in graphical or tabular format.

   Figure 27–39 shows the Performance Summary page and Metric Palette. In addition to **WebCenter Portal** performance metrics, the Metric Palette also displays general performance metrics associated with any J2EE application, for example, **ADF Application Module Pool** metrics.

   To display online help for any metric, right-click the required directory or any metric in the directory and select **Help**.

Figure 27–39 Portal Framework application - Performance Summary and Metric Palette



27.3 Customizing Key Performance Metric Thresholds and Collection

This section includes the following topics:

- Section 27.3.1, "Understanding Customization Options for Key Performance Metrics"
- Section 27.3.2, "Understanding Default Metric Collection and Threshold Settings"
- Section 27.3.3, "Configuring Thresholds for Key Metrics"
- Section 27.3.4, "Configuring Thresholds for Document Upload/Download Metrics"
- Section 27.3.5, "Configuring the Frequency of WebLogic Server Health Checks"
- Section 27.3.6, "Configuring the Number of Samples Used to Calculate Key Performance Metrics"
- Section 27.3.7, "Editing Thresholds and Collection Options for WebCenter Portal and Portal Framework Applications (metric_properties.xml)"

27.3.1 Understanding Customization Options for Key Performance Metrics

You can fine-tune how Oracle WebCenter Portal collects and reports key performance metrics to best suit your installation in several ways:

- **Customize warning thresholds for key performance metrics**

  For example, you can specify that in your installation, page response times greater then 15 seconds must trigger a warning message and report an "out-of-bounds" condition in DMS. Out-of-bound conditions also display "red" in performance charts to notify you that there is an issue.

  For more information, see:

  – Section 27.3.3, "Configuring Thresholds for Key Metrics"

  - Section 27.3.4, "Configuring Thresholds for Document Upload/Download Metrics"

- **Customize how many samples to collect for key performance metrics**

  If the default sample size (100) is too large or too small for your installation you can configure a more suitable value.

  For more informations, see Section 27.3.6, "Configuring the Number of Samples Used to Calculate Key Performance Metrics."

- **Customize health check frequency**

  If your installation demands a more aggressive schedule you can check the system health more often. The default health check frequency is 5 minutes.

  For details, see Section 27.3.5, "Configuring the Frequency of WebLogic Server Health Checks."

See also, Section 27.3.7, "Editing Thresholds and Collection Options for WebCenter Portal and Portal Framework Applications (metric_properties.xml)."

27.3.2 Understanding Default Metric Collection and Threshold Settings

You can configure metric collection options and metric threshold settings for WebCenter Portal or Portal Framework applications through the `metric_properties.xml` file. The default settings are shown in Example 27–1 and highlighted **bold**.

> **Note:** All time thresholds are specified in *milliseconds*. Memory sizes are specified in *bytes* and CPU usage is specified as a *percentage*.

Example 27–1 Default Metric Collection and Threshold Settings (metric_properties.xml)

```
<registry>
 <global_setting>
   <thread_config>
     <thread component_type="oracle_webcenter" interval="5"/>
   </thread_config>
   <health_check_config>
     <health_check name="wlsHealthCheck" enabled="true" collect="1"/>
   </health_check_config>
   <metric_config>
     <metric name="pageResponseTime" type="time" threshold="10000" comparator="gt"/>
     <metric name="portletResponseTime" type="time" threshold="10000" comparator="gt"/>
     <metric name="wlsCpuUsage" type="number" threshold="80" comparator="gt"/>
     <metric name="wlsGcTime" type="number" threshold="undef" comparator="gt"/>
     <metric name="wlsGcInvPerMin" type="number" threshold="undef" comparator="gt"/>
     <metric name="wlsActiveSessions" type="number" threshold="undef" comparator="gt"/>
     <metric name="wlsExecuteIdleThreadCount" type="number" threshold="undef" comparator="gt"/>
     <metric name="wlsActiveExecuteThreads" type="number" threshold="undef" comparator="gt"/>
     <metric name="wlsHoggingThreadCount" type="number" threshold="0" comparator="gt"/>
     <metric name="wlsOpenJdbcConn" type="number" threshold="undef" comparator="gt"/>
     <metric name="wlsHeapSizeCurrent" type="number" threshold="undef" comparator="gt"/>
    /metric_config>
   <custom_param_config>
      <custom_param name="downloadTimeThreshold" value="500"/>
      <custom_param name="downloadThroughputThreshold" value="1024"/>
      <custom_param name="uploadTimeThreshold" value="3000"/>
      <custom_param name="uploadThroughputThreshold" value="180"/>
```

```
   </custom_param_config>
  /global_setting>
</registry>
```

For descriptions of all the settings in this file, refer to the following tables:

- Table 27–37, " Key Performance Metric Threshold Configuration"
- Table 27–39, " Document Upload/Download Threshold Configuration"
- Table 27–40, " Health Check Frequency Configuration"

For information on how to modify the default settings, see "Section 27.3, "Customizing Key Performance Metric Thresholds and Collection."

27.3.3 Configuring Thresholds for Key Metrics

You can customize the default warning thresholds for some key performance metrics to make them more suitable for your Oracle WebCenter Portal installation. Table 27–36, " Configurable Metric Thresholds" lists key performance metrics you can configure and their default thresholds (if any).

Out-of-the-box, thresholds are only pre-configured for page response (*more than 10 seconds*), portlet response (*more than 10 seconds*), and CPU usage (*over 80%*).

Note: The value `undef` means that a threshold is not defined.

You can change for threshold for any of the metrics listed in Table 27–36. For example, by default, pages that take longer than 10 seconds to display trigger a warning message, report an "out-of-bounds" condition in DMS, and show "red" in performance charts to immediately notify you when page responses are too slow. Some portal applications might consider 5 seconds to be an acceptable response time, in which case you can change the threshold to 5, 000 (ms) so that your performance charts only show "red" if there really is a problem for you.

Note: You can set thresholds for document upload and download performance too. For details, Section 27.3.4, "Configuring Thresholds for Document Upload/Download Metrics."

Table 27–36 Configurable Metric Thresholds

Metric Name	Description	Default Threshold Value	Comparator
`pageResponseTime`	Number of milliseconds to render a page.	10,000 ms	gt
`portletResponseTime`	Number of milliseconds to render a portlet.	10,000 ms	gt
`wlsCpuUsage`	Percentage CPU usage of the WebLogic Server's JVM.	80%	gt
`wlsGcTime`	Average length of time (ms) the JVM spent in each run of garbage collection. The average shown is for the last five minutes.	undef	gt
`wlsGcInvPerMin`	Rate (per minute) at which the JVM is invoking its garbage-collection routine. The rate shown is for the last five minutes.	undef	gt

Table 27–36 (Cont.) Configurable Metric Thresholds

Metric Name	Description	Default Threshold Value	Comparator
wlsActiveSessions	Number of active sessions on WebLogic Server.	undef	gt
wlsExecuteIdleThreadCount	Number of execute idle threads on WebLogic Server	undef	gt
wlsActiveExecuteThreads	Number of active execute threads on WebLogic Server.	undef	gt
wlsHoggingThreadCount	Number of hogging threads on WebLogic Server.	undef	gt
wlsOpenJdbcConn	Number of open JDBC connections on WebLogic Server.	undef	gt
wlsHeapSizeCurrent	JVM's current heap size on WebLogic Server.	undef	gt

Metric thresholds are configured in `metrics_properties.xml` using the format:

```
<metric_config>
 <metric name="<metric_name>" type="<number/time/string>" threshold="<value>"
comparator="gt/lt/eq>"/>
  ...
</metric_config>
```

Table 27–36 describes each parameter.

Table 27–37 Key Performance Metric Threshold Configuration

<Metric> Parameter	Configurable	Description
name	No	Name of the metric. The metric name must exactly match the DMS sensor name as listed in Table 27–36.
type	Yes	Specifies whether the metric is a `number`, `time`, or `string`.
threshold	Yes	(Only applies when `type` is set to `number` or `time`). Specifies a numeric threshold value. If specified, you must also specify a `comparator`. For example, if portlet response times greater than 5 seconds are considered out-of-bounds: `metric name="portletResponseTime" type="time" threshold="5000" comparator="gt"` **Note:** Time must be specified in milliseconds.
comparator	Yes	Specify one of `gt`, `lt`, or `eq`. Where: `gt` - greater than `lt` - less than `eq` - equal to

To edit one or more metric thresholds, follow the steps in Section 27.3.7, "Editing Thresholds and Collection Options for WebCenter Portal and Portal Framework Applications (metric_properties.xml)."

27.3.4 Configuring Thresholds for Document Upload/Download Metrics

Oracle WebCenter Portal provides several metric parameters that enable you to configure warning thresholds for document upload and download performance in WebCenter Portal or your Portal Framework application (see Table 27–38).

Table 27–38 Configurable Document Upload/Download Metric Thresholds

Metric Name	Description	Default Value
`downloadTimeThreshold`	Acceptable download time for a document (in milliseconds): ▪ Documents that download faster than or equal to the time specified by the `downloadTimeThreshold` are considered acceptable (always show "green" in performance charts). ▪ Document downloads that take longer than the time are only considered acceptable (green) if the download rate is better than the `downloadThroughputThreshold`. Document downloads that exceed the download time threshold and fail to meet the throughput threshold fall below the expected performance criteria and generate an out-of-bound condition (show "red" in performance charts). The default download time threshold is 500ms (0.5 seconds).	500 ms
`downloadThroughputThreshold`	Acceptable download rate for a document (in Kilobytes per second). This threshold is only applied to document downloads that exceed the `downloadTimeTreshold`.	1024 KB/sec
`uploadTimeThreshold`	Acceptable upload time for a document (in milliseconds). The default upload time threshold is 3000ms (3 seconds), that is, six times longer than the default download time.	3000 ms
`uploadThroughputThreshold`	Acceptable upload rate for a document (in Kilobytes per second). This threshold is only applied to document uploads that exceed the `uploadTimeThreshold`.	180 KB/sec

If your users expect faster upload/download speeds you might want to reduce the default time thresholds or increase the throughput thresholds provided.

Document upload and download thresholds are configured in `metrics_properties.xml` using the format:

```
<custom_param_config>
   <custom_param name="<document_threshold_name>" value="<value>"/>
   ...
</custom_param_config>
```

Table 27–39 describes each parameter.

Table 27–39 Document Upload/Download Threshold Configuration

custom_param Parameter	Configurable	Description
`document_threshold_name`	No	Name of the customizable metric parameter. The names must exactly match the DMS sensor name as listed in Table 27–38.

Table 27–39 (Cont.) Document Upload/Download Threshold Configuration

custom_param Parameter	Configurable	Description
`value`	Yes	Value of the customizable metric parameter. For example, to set a 300ms download time threshold for documents: `<custom_param name="downloadTimeThreshold" value="300"/>`

To edit one or more metric thresholds, follow the steps in Section 27.3.7, "Editing Thresholds and Collection Options for WebCenter Portal and Portal Framework Applications (metric_properties.xml)."

27.3.5 Configuring the Frequency of WebLogic Server Health Checks

Out-of-the-box, the general health of the WebLogic Server on which WebCenter Portal or your Portal Framework application is deployed is checked every 5 minutes and the results are reported on the "Understanding WebLogic Server Metrics" page.

If your installation demands a more aggressive schedule you can check the system health more often.

Health check frequency is configured in `metrics_properties.xml` using the format:

```
<thread_config>
 <thread component_type="oracle_webcenter" interval="<value>"/>
</thread_config>
```

Table 27–40 describes each parameter.

Table 27–40 Health Check Frequency Configuration

<thread> Parameter	Default Value	Configurable	Description
`component_type`	`oracle_webcenter`	No	For Oracle WebCenter Portal, the component_type is always `oracle_webcenter`.
`interval`	`5 minutes`	Yes	Specifies the interval between health checks, in minutes. For example: `<thread component_type="oracle_webcenter" interval="10"/>`

To change the frequency, follow the steps in Section 27.3.7, "Editing Thresholds and Collection Options for WebCenter Portal and Portal Framework Applications (metric_properties.xml)."

27.3.6 Configuring the Number of Samples Used to Calculate Key Performance Metrics

Oracle WebCenter Portal collects and reports recent performance for several key performance metrics (page, portlet, document, and WebLogic Server) based on a fixed number of data samples. Out-of-the-box, the last *100* samples of each metric type are used to calculate these key performance metrics, that is, 100 samples for page metrics, 100 samples for portlet metrics, and so on.

You can increase or decrease the sample set to suit your installation. If you decide to increase the number of samples you must consider the additional memory cost of doing so, since all the key performance metrics samples are maintained in memory. Oracle recommends that you specify a few hundred at most. See Section 27.1.1, "Understanding Oracle WebCenter Portal Metric Collection."

> **Note:** Since all "out-of-bounds" metrics are recorded in the managed server's diagnostic log, you can always scan the logs at a later date or time to see what happened in the past, that is, beyond the 'N' metric samples that are temporarily held in memory.

The server startup property `WC_HEALTH_MAX_COLLECTIONS` determines the number of metric samples collected by Oracle WebCenter Portal. If the property is not specified, 100 samples are collected.

To customize the number of samples collected for key performance metrics:

1. Log in to WebLogic Server Administration Console.
2. Navigate to the managed server on which your application is deployed.

   For WebCenter Portal, navigate to **Environment> Servers> WC_Spaces**.

   For Portal Framework applications, navigate to **Environment> Servers> <name of the custom managed server>**, for example, WC_CustomPortal.
3. Click the **Server Start** tab.
4. In the **Arguments** text area, enter the server startup argument `WC_HEALTH_MAX_COLLECTIONS` and specify the number of samples you want to collect.

   For example:

   ```
   -DWC_HEALTH_MAX_COLLECTIONS=200
   ```

   Separate multiple arguments with a space. For example:

   ```
   -DWC_HEALTH_MAX_COLLECTIONS=200
   -DWEBCENTER_METRIC_PROPERTIES=/scratch/mythresholds/metric_properties.xml
   ```
5. Restart the managed server.

27.3.7 Editing Thresholds and Collection Options for WebCenter Portal and Portal Framework Applications (metric_properties.xml)

To change metric thresholds and collection criteria for WebCenter Portal or Portal Framework applications:

1. Copy the XML snippet in Example 27–1, "Default Metric Collection and Threshold Settings (metric_properties.xml)" and save it to a text file named `metric_properties.xml`.
2. Edit metric collection parameters and/or metric thresholds in `metric_properties.xml`, as required.

> **Note:** You must consider your machine resources, as well as the system topology and configuration when choosing suitable thresholds for your Oracle WebCenter Portal installation. As each installation is different, most metrics do not have default or recommended threshold settings.

A description of all the settings and their defaults (if any) are described in the following tables:

- Table 27–37, " Key Performance Metric Threshold Configuration"
- Table 27–40, " Health Check Frequency Configuration"
- Table 27–39, " Document Upload/Download Threshold Configuration"

3. Copy the updated `metric_properties.xml` file to:
   - Your *DOMAIN_HOME*.
   - Another suitable directory.
4. Configure the server startup argument `WEBCENTER_METRIC_PROPERTIES` to point to the full path of the properties file:
   a. Log in to WebLogic Server Administration Console.
   b. Navigate to the managed server on which your application is deployed.

      For WebCenter Portal, navigate to **Environment> Servers> WC_Spaces**.

      For Portal Framework applications, navigate to **Environment> Servers> <name of the custom managed server>**, for example, WC_CustomPortal.
   c. Click the **Server Start** tab.
   d. In the **Arguments** text area, enter the `WEBCENTER_METRIC_PROPERTIES` argument and specify the full path of the properties file.

      For example:

      ```
      -DWEBCENTER_METRIC_PROPERTIES=/scratch/mythresholds/metric_properties.xml
      ```

      > **Notes:** If you only specify the file name, Oracle WebCenter Portal looks for this file in your *DOMAIN_HOME*.
      >
      > Separate multiple arguments with a space. For example:
      >
      > ```
      > -DWC_HEALTH_MAX_COLLECTIONS=200
      > -DWEBCENTER_METRIC_PROPERTIES=/scratch/mythresholds/metric_properti
      > es.xml
      > ```

   e. Restart the managed server.

27.4 Diagnosing and Resolving Performance Issues with Oracle WebCenter Portal

The performance metrics described in this chapter enable you to quickly assess the current status and performance of WebCenter Portal or your Portal Framework application from Fusion Middleware Control. When performance is slow, further investigations may be required for you to fully diagnose and fix the issue. For

guidance, see Section 27.1.3, "Using Key Performance Metric Data to Analyze and Diagnose System Health."

Some common performance issues and actions are described in this chapter:

- Section 27.1.4, "Understanding Some Common Performance Issues and Actions"
- Section 27.1.12.3, "Troubleshooting Common Issues with Tools and Services"

For more detailed troubleshooting tips relating to performance, see Appendix G, "Troubleshooting Oracle WebCenter Portal."

27.5 Tuning Oracle WebCenter Portal Performance

See the "Oracle WebCenter Portal Performance Tuning" chapter in the *Oracle Fusion Middleware Performance and Tuning Guide* for information on tuning WebCenter Portal and Portal Framework applications. For example, how to tune the system limit (open-files-limit), JDBC data sources, JVM arguments, session timeouts, page timeouts, connection timeouts, concurrency timeouts, caching, and more.

28

Managing Oracle WebCenter Portal Logs

This chapter describes diagnostic logging in WebCenter Portal.

This chapter includes the following topics:

- Section 28.1, "Introduction to Diagnostic Logging"
- Section 28.2, "Viewing and Configuring Log Information"

> **Permissions:** To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin`, `Operator`, or `Monitor` role through the Oracle WebLogic Server Administration Console.
>
> See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

28.1 Introduction to Diagnostic Logging

All diagnostic information relating to startup and shutdown information, errors, warning messages, access information on HTTP requests, and other additional information is stored in log files.

For general information about managing and analyzing logs using Fusion Middleware Control and WLST, see the "Managing Log Files and Diagnostic Data" chapter in *Oracle Fusion Middleware Administrator's Guide*.

See also, the "Understanding the Diagnostic Framework" section in *Oracle Fusion Middleware Administrator's Guide*.

WebCenter Portal Diagnostics Log

The diagnostics log file for WebCenter Portal is `WC_Spaces-diagnostic.log`.

This log is available under the *`DOMAIN_HOME`*`/servers/WC_Spaces/logs` directory.

Portal Framework Application Diagnostics Log

The diagnostics log file for Portal Framework applications is available in the *`DOMAIN_HOME`*`/servers/`*`ServerName`*`/logs` directory.

The log file follows the naming convention: *`ServerName`*`-diagnostic.log`

For example, if the Portal Framework application is deployed on a managed server named, `WC_CustomPortal`, diagnostics log information are stored under the *`DOMAIN_HOME`*`/servers/WC_CustomPortal/logs` directory, and the log file name is `WC_CustomPortal-diagnostic.log`.

Oracle WebCenter Portal Message IDs

Oracle WebCenter Portal log messages fall into these categories:

Table 28–1 Oracle WebCenter Portal Message Categories

Message ID Range	Message Category
BI Integration	WCS-01001 ~ WCS-02000
Blogs	WCS-02001 ~ WCS-03000
Calendar Tasks	WCS-03001 ~ WCS-04000
Collaboration Integration	WCS-04001 ~ WCS-05000
Portal Builder	WCS-05001 ~ WCS-06000
VCR	WCS-06001 ~ WCS-07000
Document Library	WCS-07001 ~ WCS-08000
Discussions	WCS-08001 ~ WCS-09000
Mail	WCS-09001 ~ WCS-10000
Explorer Toolbar	WCS-10001 ~ WCS-11000
Desktop Integration	WCS-11001 ~ WCS-12000
Lifecycle	WCS-12001 ~ WCS-13000
Links	WCS-13001 ~ WCS-14000
Lists	WCS-14001 ~ WCS-15000
Navigation	WCS-15001 ~ WCS-16000
Page Editor	WCS-16001 ~ WCS-17000
Page Templates	WCS-17001 ~ WCS-18000
People	WCS-18001 ~ WCS-19000
Personal WebCenter	WCS-19001 ~ WCS-20000
Provisioned Apps	WCS-20001 ~ WCS-21000
Ratings / Comments	WCS-21001 ~ WCS-22000
Region	WCS-22001 ~ WCS-23000
Resource Catalog	WCS-23001 ~ WCS-24000
Rich Text Editor	WCS-24001 ~ WCS-25000
Roles	WCS-25001 ~ WCS-26000
Search	WCS-26001 ~ WCS-27000
Skins	WCS-27001 ~ WCS-28000
Smart Tags	WCS-28001 ~ WCS-29000
Subscription	WCS-29001 ~ WCS-30000
Wiki	WCS-30001 ~ WCS-31000
WebCenter Portal Editor	WCS-31001 ~ WCS-32000
Worklist	WCS-32001 ~ WCS-33000
Recent Activities	WCS-33001 ~ WCS-34000
Content Adapters	WCS-34001 ~ WCS-35000
VCR ADF Integration	WCS-35001 ~ WCS-36000

Table 28–1 (Cont.) Oracle WebCenter Portal Message Categories

Message ID Range	Message Category
Pages	WCS-36001 ~ WCS-37000
Notes	WCS-37001 ~ WCS-38000
RSS	WCS-38001 ~ WCS-39000
Portlet Binding	WCS-39001 ~ WCS-40000
Portlet Runtime	WCS-40001 ~ WCS-41000
DesignTime@Runtime	WCS-41001 ~ WCS-42000
External Application	WCS-42001 ~ WCS-43000
Service Framework	WCS-43001 ~ WCS-44000
Security Framework	WCS-44001 ~ WCS-45000
Portlet Design-Time	WCS-45001 ~ WCS-46000
Resource Catalog Viewer	WCS-46001 ~ WCS-47000
People Connections	WCS-47001 ~ WCS-48000
Preferences	WCS-48001 ~ WCS-49000
REST	WCS-49001 ~ WCS-50000
Notifications	WCS-50001 ~ WCS-51000
Office integration	WCS-51001 ~ WCS-52000
Blogs	WCS-52001 ~ WCS-53000
Activity Graph	WCS-53001 ~ WCS-54000
VCR (from WLP	WCS-54001 ~ WCS-55000
WebCenter Content SPI	WCS-55001 ~ WCS-56000
Personalization (wcps-util)	WCS-56001 ~ WCS-57000
Personalization (wcps-cmis-provider	WCS-57001 ~ WCS-58000
Decision Component Integration (wcps-decision	WCS-58001 ~ WCS-59000
Conductor (wcps-conductor)	WCS-59001 ~ WCS-60000
Properties Provider	WCS-60001 ~ WCS-61000
RESTClient	WCS-61001 ~ WCS-62000
Translations	WCS-62001 ~ WCS-63000
Analytics	WCS-63001 ~ WCS-64000
JAX-RS Framework	WCS-64001 ~ WCS-65000
Data Presenter	WCS-65001 ~ WCS-66000
Knowledge Directory	WCS-66001 ~ WCS-67000
Concurrency Package	WCS-67001 ~ WCS-68000
PortalApps Integration	WCS-68001 ~ WCS-69000
System Management	WCS-69001 ~ WCS-70000
Performance Out-of-bounds	WCS-69201 ~ WCS-70000

Table 28–1 (Cont.) Oracle WebCenter Portal Message Categories

Message ID Range	Message Category
Nitrous	WCS-70001 ~ WCS-71000

Out-Of-Bound Conditions for Oracle WebCenter Portal Performance Metrics

Out-of-bound conditions are also logged in managed server diagnostic logs so you can examine historical events at any time. Performance related messages are logged with the message ID prefix `WCS-692<nn>`and include the metric name, the value, and a message describing the metric that is out-of-bounds.

Here are some examples of messages that you might see in diagnostic logs for WebCenter Portal:

```
[WC_Spaces] [WARNING] [WCS-69251] [oracle.webcenter.system-management] [tid:
 [ACTIVE].ExecuteThread: '4' for queue: 'weblogic.kernel.Default (self-tuning)']
 [userId: weblogic] [ecid:
 6356ef0164cbad47:3fe105c5:13b4e847973:-8000-0000000000000031,0] [APP:
 webcenter#11.1.1.4.0] [DSID: 0000JhEYRT^EgKG_Ix8Dyf1Ghz32000005]
 pageResponseTime: 22223 ms of PersonalSpace/Activities is out-of-bounds

[WC_Spaces] [WARNING] [WCS-69252] [oracle.webcenter.system-management] [tid:
 oracle.webcenter.DefaultTimer] [ecid: 0000JhEX92mEgKG_Ix8Dyf1Ghz32000002,0] [APP:
webcenter#11.1.1.4.0]
wlsCpuUsage: 21.92100394175851 % of WebLogicServer is out-of-bounds

[WC_Spaces] [WARNING] [WCS-69255] [oracle.webcenter.system-management] [tid:
 [ACTIVE].ExecuteThread: '0' for queue: 'weblogic.kernel.Default (self-tuning)']
 [userId: weblogic] [ecid:
 6356ef0164cbad47:3fe105c5:13b4e847973:-8000-0000000000000060,0] [APP:
 webcenter#11.1.1.4.0] [DSID: 0000JhEYRT^EgKG_Ix8Dyf1Ghz32000005]
 downloadThroughput: 11.63793103448276 KB/sec of 3209 is out-of-bound

[WC_Spaces] [WARNING] [WCS-69253] [oracle.webcenter.system-management] [tid:
 pool-3-daemon-thread-1] [userId: weblogic] [ecid:
 6356ef0164cbad47:3fe105c5:13b4e847973:-8000-0000000000000088,0:16] [APP:
 webcenter#11.1.1.4.0] portletResponseTime: 20523 ms of Portlet:
slowRenderingPortlet from Web Producer pkjpdk is out-of-bounds
```

28.2 Viewing and Configuring Log Information

This section includes the following topics:

- Section 28.2.1, "Viewing and Configuring WebCenter Portal Logs"
- Section 28.2.2, "Viewing and Configuring Portal Framework Application Logs"

28.2.1 Viewing and Configuring WebCenter Portal Logs

To view log messages for a WebCenter Portal application:

1. In Fusion Middleware Control Console, navigate to the home page for WebCenter Portal.

   See Section 6.2, "Navigating to the Home Page for WebCenter Portal."

2. From the **WebCenter Portal** menu, select **Logs** > **View Log Messages**.
3. In the **Log Messages** page, search for warnings, errors, notifications, and so on.

To configure log files for WebCenter Portal:

1. In Fusion Middleware Control Console, navigate to the home page for WebCenter Portal.

   See Section 6.2, "Navigating to the Home Page for WebCenter Portal."

2. From the **WebCenter Portal** menu, select **Logs** > **Log Configuration**.
3. In the **Log Configuration** page, in the **Log Files** tab, configure log settings.

For more information, see the "Viewing and Searching Log Files" section in *Oracle Fusion Middleware Administrator's Guide*.

28.2.2 Viewing and Configuring Portal Framework Application Logs

To view log messages for Portal Framework applications:

1. In Fusion Middleware Control Console, navigate to the home page for the Portal Framework application.

   See Section 6.3, "Navigating to the Home Page for Portal Framework Applications."

2. From the **Application Deployment** menu, select **Logs > View Log Messages**.
3. In the **Log Messages** page, search for warnings, errors, notifications, and so on.

To configure log files for Portal Framework applications:

1. In Fusion Middleware Control Console, navigate to the home page for the Portal Framework application.

   See Section 6.3, "Navigating to the Home Page for Portal Framework Applications."

2. From the **Application Deployment** menu, select **Logs > Log Configuration**.
3. In the **Log Configuration** page, in the **Log Files** tab, configure log settings.

For more information, see the "Viewing and Searching Log Files" section in *Oracle Fusion Middleware Administrator's Guide*.

29

Managing Oracle WebCenter Portal Audit Logs

This chapter provides an introduction to managing audit logging for WebCenter Portal.

This chapter includes the following sections:

- Section 29.1, "Introduction to Managing Audit Logs"
- Section 29.2, "Configuring Audit Logging"
- Section 29.3, "Viewing WebCenter Portal Audit Events"

> **Permissions:** To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console. Users with the `Monitor` or `Operator` roles can view security information but cannot make changes.
>
> See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

29.1 Introduction to Managing Audit Logs

When enabled, audit logging tracks Portal-related events as part of the Fusion Middleware Audit Service. Audit log events are stored in a file (the Audit Bus-stop) by default, but can also be uploaded to a database for persistency (for more information, see Section 29.2.2, "Configuring the Audit Store Database"). The Audit Bus-stop file has a limited capacity so storing log information in a database where events can be queried long after their occurrence is recommended.

> **Note:** If you enable WebCenter Portal Impersonation, it is highly recommended that you also enable audit logging. When Impersonation is enabled, audit logging tracks the impersonator, impersonatee, and the context surrounding an event.

Audit logging provides the following key benefits:

- Events that alter the security settings of Portal, Portal Server, and major Portal Server artifacts are traceable
- Definable logging levels

- Events logged are available in perpetuity when uploaded to a database
- Reports on audit events are available through the Audit Service

For more information about the Audit Service and configuring the Audit Service, see "Introduction to Oracle Fusion Middleware Audit Framework" in the *Oracle Fusion Middleware Application Security Guide*. For information about configuring the Audit Service to use a database, see "Configuring and Managing Auditing" in the *Oracle Fusion Middleware Application Security Guide*. For information about out-of-the-box Audit Service reports, see "Pre-built Audit Reports" in the *Oracle Fusion Middleware Application Security Guide*.

29.2 Configuring Audit Logging

This section describes how to turn logging on and off for WebCenter Portal, how to set the log level, and how to set up the Audit Store Database.

This section contains the following subsections:

- Section 29.2.1, "Setting the Logging Level"
- Section 29.2.2, "Configuring the Audit Store Database"

29.2.1 Setting the Logging Level

By default, audit logging for WebCenter Portal is turned off (that is, set to `None`). To turn it on, set the logging level to a value other than `None` (for example, `Low`) as shown in the examples below. For the details of which logging categories are included for each logging level, see Section 29.3.1, "Using WebCenter Portal Audit Logs."

Use the following WLST commands to modify the audit logging level for WebCenter Portal audit events:

To set the logging level to `Low`:

```
setAuditPolicy(componentType="webcenter#11.1.1.4.0",filterPreset="Low")
```

Set the logging level to `Medium`:

```
setAuditPolicy(componentType="webcenter#11.1.1.4.0",filterPreset="Medium")
```

To turn logging off for WebCenter Portal:

```
setAuditPolicy(componentType="webcenter#11.1.1.4.0",filterPreset="None")
```

Successful execution does not throw any error and completes silently. Restart the `WC_Spaces` server to complete the logging level change.

For information about additional WLST commands you can use to manage and configure audit logging, see "WLST Commands for Auditing" in the *Oracle Fusion Middleware Application Security Guide*.

29.2.2 Configuring the Audit Store Database

The audit store is a database that contains a pre-defined Oracle Fusion Middleware Audit Framework schema created by the Repository Creation Utility (RCU). By default, audit logs are stored as files in the `auditlogs` directory as shown in the following example:

```
DOMAIN_HOME/servers/WC_Spaces/logs/auditlogs/webcenter#11.1.1.4.0/audit_1_0.log
```

Once database persistence has been configured, the Audit loader picks up data from this file and puts it in the Audit Framework schema. For information about configuring the Audit Service to use a database, see "Configuring and Managing Auditing" in the *Oracle Fusion Middleware Application Security Guide*.

You will need to know the name of the audit schema (the suffix is always IAU). You will also need to set the audit repository to the database as shown below:

```
setAuditRepository(switchToDB='true',dataSourceName='jdbc/AuditDB',interval='15')
```

Note: The audit data in the store is expected to be cumulative and will grow over time. Ideally, the database should not be an operational database used by any other applications, and should be a standalone RDBMS used for audit purposes only.

29.3 Viewing WebCenter Portal Audit Events

This section describes the WebCenter Portal Impersonation events that are available in the audit log, and shows a simple SQL statement that you can use to query the audit schema for impersonation events.

This section includes the following subsections:

- Section 29.3.1, "Using WebCenter Portal Audit Logs"
- Section 29.3.2, "Querying the Audit Schema"

29.3.1 Using WebCenter Portal Audit Logs

Table 29–1 lists the WebCenter Portal audit events that appear in the audit log depending on the log level that is set. The various WebCenter Portal tools (such as documents, announcements, discussions, wiki and blog, forum, forum message, forum topic, forum category) are identified in the log by their corresponding ToolArtifactID and ToolType.

When the log level is set to `Low`, events in the following categories will be logged:

- PortalLifeCycle
- PortalRoleManagement
- PortalRoleMemberManagement
- PortalToolAccessManagement
- ImpersonationSessionMgmt

When the log level is set to `Medium`, events in the following additional categories will be logged:

- PortalToolsManagement
- PortalPagesManagement

Table 29–1 WebCenter Portal Audit Events

Event Category	Event Name	Event Payload
PortalLifeCycle	LoginPortalServer, CreatePortal, DeletePortal, ImportPortal, ExportPortal, DeployPortal, PropagatePortal	InitiatorUID, InitiatorMail, InitiatorDisplayName, ImpersonatorUID, PortalID, PortalName, PotalDisplayName, PortalURL, PortalTemplate, PortalOldState, PortalNewState, TargetPortalConnection
PortalRoleManagement	CreateRole DeleteRole PermissionUpdate	InitiatorUID, InitiatorMail, InitiatorDisplayName, ImpersonatorUID, PortalID, PortalName, RoleName, RoleTemplate, PermissionClass, PermissionName, PermissionActionsGranted, PermissionActionsRevoked
PortalRoleMemberManagement	AddMemberToRole RemoveMemberFromRole	InitiatorUID, InitiatorMail, InitiatorDisplayName, ImpersonatorUID, PortalID, PortalName, RoleName, MemberType, MemberUID, ServiceID
ImpersonationSessionMgmt	GrantImpersonationAccess RevokeImpersonationAccess BeginImpersonation EndImpersonation	InitiatorUID, InitiatorMail, InitiatorDisplayName, ImpersonatorUID, ImpersonateeUID, PortalID, PortalName, ImpersonationStartTime, ImpersonationEndTime, ImpersonationGrantStartTime, ImpersonationEndTime, ImpersonationRightRevokeTime
PortalToolsManagement	CreateTool, DeleteTool ModifyTool	InitiatorUID, InitiatorMail, InitiatorDisplayName, ImpersonatorUID, PortalID, PortalName, ToolArtifactID, ToolName, ToolType
PortalToolAccessManagement	ToolAccessPermissionUpdate GrantToolAccess RevokeToolAccess	InitiatorUID, InitiatorMail, InitiatorDisplayName, ImpersonatorUID, PortalID, PortalName, ToolName, ToolType, ToolArtifactID, MemberUID, MemberType, PermissionActionsGranted, PermissionActionsRevoked, PermissionClass, PermissionName
PortalPagesManagement	CreatePage DeletePage	InitiatorUID, InitiatorMail, InitiatorDisplayName, ImpersonatorUID, PortalID, PortalName, PageID, PageName

29.3.2 Querying the Audit Schema

Once you've configured the audit schema and the audit repository is set to database, you can create reports based on this generated audit data. Follow the steps below to create a report:

1. Generate a view based on audit tables by running the following command to generate a SQL file that can then be used to create a view for the WebCenter Portal component-specific data from audit DB tables:

   ```
   createAuditDBView(fileName="/tmp/WCPortalAuditView.sql",
   componentType="webcenter#11.1.1.4.0")
   ```

   The IAU schema owner (for example, `TEST_IAU`) will need to have 'create view' privileges. To create the view, run the `WCPortalAuditView.sql` file or run the following SQL command as a system DBA:

   ```
   grant create view to TEST_IAU
   ```

The created view will have name like 'webcenter#11_1_1_4_0_AUDITVIEW'.

2. Use the view to query the audit database using WebCenter Portal tool audit attribute names as table column name as shown in the following examples. Open the `WCPortalAuditView.sql` file to see the mapping of table column names with WebCenter Portal attributes.

   - The following SQL statement returns all the attributes of WebCenter Portal tools that are logged with the event types `BeginImpersonation` and `EndImpersonation`:

     ```
     select * from webcenter#11_1_1_4_0_AUDITVIEW where EventType like
     '%Impersonation';
     ```

   - The following SQL statement lists all users who have deleted any portal along with the deleted portal information:

     ```
     select InitiatorUID,InitiatorMail,PortalID,PortalName,PortalURL from
     webcenter#11_1_1_4_0_AUDITVIEW where EventType = 'DeletePortal';
     ```

   - The following SQL statement returns all audit data for WebCenter Portal:

     ```
      select * from webcenter#11_1_1_4_0_AUDITVIEW;
     ```

If you want to regularly monitor WebCenter Portal activities you can create a SQL data control using SQL queries and drop the data control as a table or other visualization onto a portal page. For more information about SQL data controls, see "Working with Data Presenter" in the *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Part VII

Security

This part of *Oracle Fusion Middleware Administering Oracle WebCenter Portal* presents security administration topics for Oracle WebCenter Portal.

Part VII contains the following chapter:

30

Managing Oracle WebCenter Portal Security

This chapter provides an introduction to securing WebCenter Portal and Portal Framework applications, and describes the security configuration that is in place when applications are initially deployed. This chapter also includes a troubleshooting section that provides solutions for common security-related configuration issues.

This chapter includes the following sections:

- Section 30.1, "Introduction to Application Security"
- Section 30.2, "Default Security Configuration"
- Section 30.3, "Troubleshooting Security Configuration Issues"

For information about specific aspects of configuring security for WebCenter Portal and Portal Framework applications, see:

- Chapter 31, "Configuring the Identity Store"
- Chapter 32, "Configuring the Policy and Credential Store"
- Chapter 33, "Configuring Single Sign-on"
- Chapter 34, "Configuring Portal Framework Applications for Single Sign-on"
- Chapter 35, "Configuring SSL"
- Chapter 36, "Configuring WS-Security"
- Chapter 37, "Configuring Security for Portlet Producers"
- Chapter 43, "Administering Portal Framework Applications Using the Administration Console"

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console. Users with the `Monitor` or `Operator` roles can view security information but cannot make changes.

See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

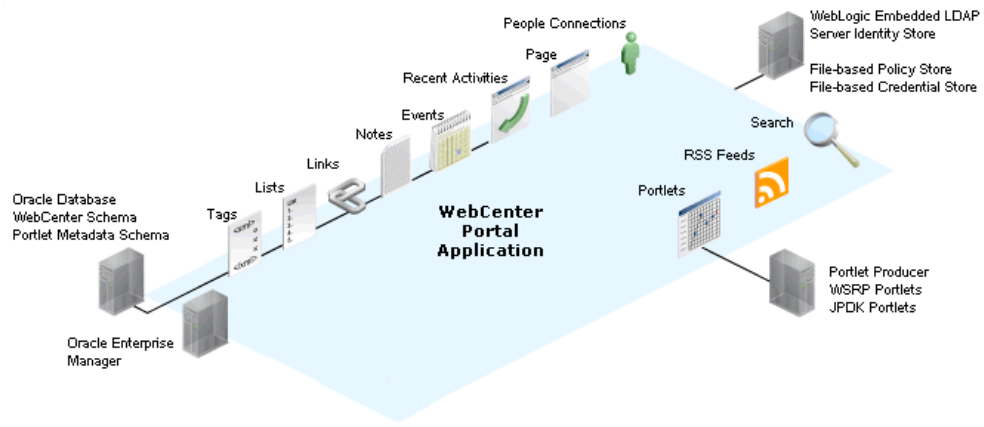
30.1 Introduction to Application Security

The recommended security model for WebCenter Portal and Portal Framework applications is based on Oracle ADF Security, which implements the Java Authentication and Authorization Service (JAAS) model. For more information about

Oracle ADF Security, see *Oracle Fusion Middleware Fusion Developer's Guide for Oracle Application Development Framework*.

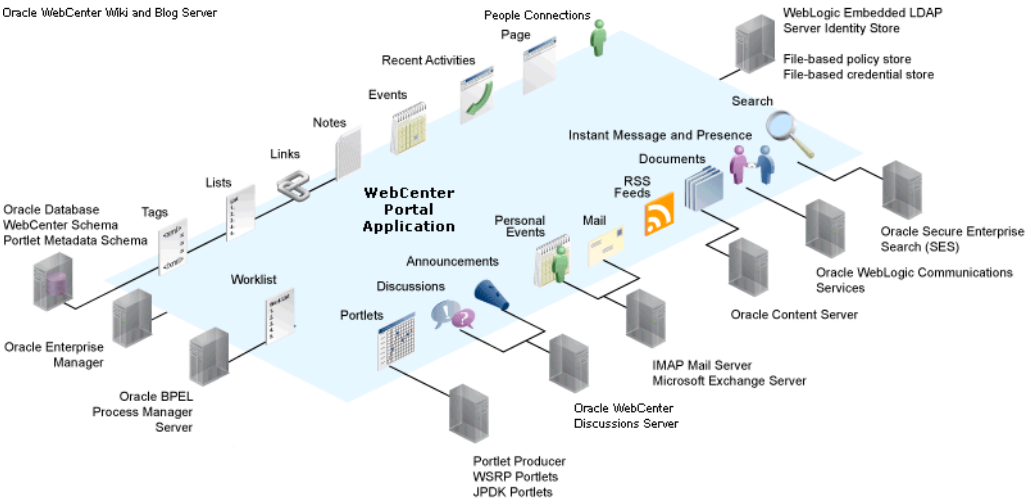
Figure 30–1 shows the relationship between a WebCenter Portal or Portal Framework application deployment and its services, servers, portlets, portlet producers, its identity, credential and policy stores, and Oracle Enterprise Manager.

Figure 30–1 Basic WebCenter Portal Application Architecture



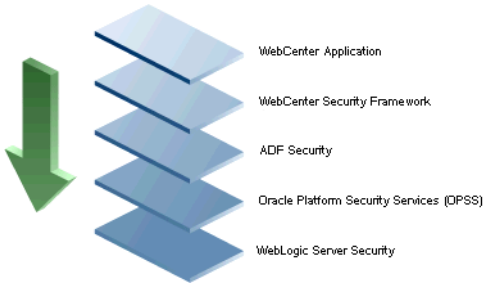
The diagram in Figure 30–2 shows a basic WebCenter Portal or Portal Framework application after deployment with its back-end server connections.

Figure 30–2 WebCenter Portal Application Architecture with Back-end Server Connections



The diagram in Figure 30–3 shows the security layers for WebCenter Portal and Portal Framework applications.

Figure 30–3 WebCenter Portal Security Layers



WebCenter Portal and Portal Framework applications share the same four bottom security layers (WebCenter Security Framework, ADF Security, OPSS, and WebLogic Server Security). The application layer will, of course, depend on the implementation.

WebCenter Portal Application Security

WebCenter Portal provides support for:

- Application role management and privilege mapping
- Self-registration
- Portal-level security management
- External application credential management

WebCenter Portal Security Framework

The WebCenter Portal Security Framework provides support for:

- Service Security Extension Framework (a common permission-based and role-mapping based model for specifying the security model for services)
- Permission-based authorization
- Role-mapping based authorization
- External applications and credential mapping

ADF Security

ADF Security provides support for:

- Page authorization
- Task flow authorization
- Secure connection management
- Credential mapping APIs
- Logout invocation, including logout from SSO-enabled configurations with Oracle Access Manager and Oracle SSO
- Secured login URL for ADF Security-based applications (the adfAuthentication servlet)

Oracle Platform Security Services (OPSS)

OPSS provides support for:

- Anonymous-role
- Authenticated-role

- Identity store, policy store, and credential store
- Identity Management Services
- Oracle Web Service Manager Security
- Authorization
- Policy and Credential Lifecycle

WebLogic Server Security

WebLogic Server Security provides support for:

- WebLogic authenticators
- Identity asserters
- J2EE container security
- SSL

30.2 Default Security Configuration

This section describes the security configuration that is in place when WebCenter Portal and Portal Framework applications are deployed, and the configuration tasks that should be carried out after deployment:

- Section 30.2.1, "Administrator Accounts"
- Section 30.2.2, "Application Roles and Enterprise Roles"
- Section 30.2.3, "Default Identity and Policy Stores"
- Section 30.2.4, "Default Policy Store Permissions and Grants"
- Section 30.2.5, "Post-deployment Security Configuration Tasks"

30.2.1 Administrator Accounts

Portal Framework applications do not contribute any pre-seeded accounts, and therefore rely on the system administrator account (`weblogic` by default) that is set up when Oracle Fusion Middleware is installed. Use this administrator account to log into Fusion Middleware Control and set up new accounts.

Although the WebCenter Portal application does not contribute any pre-seeded accounts, there are certain pre-seeded grants that are given to the default system administrator account (`weblogic`) for the WebCenter Portal application. If your installation does not use `weblogic` as the account name for the system administrator role, you must configure one or more other users for this role as described in Section 32.6, "Managing Users and Application Roles."

Note: The `weblogic` account is a system administrator account and should not be used to create user-level artifacts. The `weblogic` account should only be used to create new user accounts in Fusion Middleware Control.

30.2.2 Application Roles and Enterprise Roles

Application roles differ from roles that appear in the identity store portion of the embedded LDAP server or in roles defined by the enterprise LDAP provider.

Application roles are specific to an application and defined in an application-specific stripe of the policy store.

Enterprise roles, which are stored in the enterprise identity store, apply at the enterprise level. That is, the roles and permissions that you or a system administrator define within the enterprise identity store do not imply permissions within an application.

Within WebCenter Portal or a Portal Framework application you can assign application roles and permissions to users in the corporate identity store. You can also assign application roles and permissions to enterprise roles defined in the enterprise identity store.

30.2.3 Default Identity and Policy Stores

By default, WebCenter Portal and Portal Framework applications are configured to use a file-based embedded LDAP identity store to store application-level user IDs, and a file-based LDAP policy store to store policy grants.

Although secure, the embedded LDAP identity store is not a "production-class" store and should be replaced with an external LDAP-based identity store such as Oracle Internet Directory for enterprise production environments. For a list of supported identity store LDAP servers, see the "Supported LDAP Identity Store Types" section in the *Oracle Fusion Middleware Application Security Guide*.

> **Caution:** The default file-based policy store should only be used for development, and only for single-node WebCenter Portal configurations. For enterprise deployments you must reassociate the policy and credential store with a database, or with an external LDAP-based store as described in Chapter 31, "Configuring the Identity Store."

The policy and credential stores can use either Oracle Internet Directory 11gR1 or 10.1.4.3, or Oracle RDBMS (releases 10.2.0.4 or later; releases 11.1.0.7 or later; and releases 11.2.0.1 or later). Note that when using an external LDAP-based store, the policy and credential stores must use the same LDAP server. Similarly, when using a database, the policy and credential stores must use the same database.

For more information about the supported identity store and policy and credential store configurations, see the "Supported LDAP-, DB-, and File-Based Services" section in the *Oracle Fusion Middleware Application Security Guide*. For more information on reconfiguring the identity store and the policy and credential stores, see Chapter 31, "Configuring the Identity Store"and Chapter 32, "Configuring the Policy and Credential Store."

Note: By default, discussions are configured to use the embedded LDAP identity store: All users in the embedded LDAP store can log onto the discussions server, and all users in the `Administrators` group have administrative privileges on the discussions server.

If you reassociate the identity store with an external LDAP server, you must either move the system administrator account to the external LDAP (as described in Section 31.4, "Moving the Administrator Account to an External LDAP Server"), or if you choose not to move the administrator account, you must perform some additional steps to identify the new administrator account for the discussions server as described in Section 31.4.1, "Migrating the Discussions Server to Use an External LDAP."

For WebCenter Portal, both the WebCenter Portal application and Content Server must share the same LDAP server. For more information, see Section 31.5, "Configuring the Oracle Content Server to Share the WebCenter Portal Identity Store LDAP Server."

30.2.3.1 File-based Credential Store

The out-of-the-box credential store is wallet-based (that is, file-based) and is contained in the file `cwallet.sso`. The location of this file is specified in the Oracle Platform Security configuration file `jps-config.xml`. When you reassociate the policy store to an LDAP directory, the application credentials are automatically migrated to the same LDAP directory as the policy store.

30.2.4 Default Policy Store Permissions and Grants

The ADF Security permissions model supports both permission-based and role-based authorization. These two types of authorization, and the default Policy Store permissions and code based grants are discussed in the following sections:

- Section 30.2.4.1, "Permission-based Authorization"
- Section 30.2.4.2, "Role-mapping Based Authorization"
- Section 30.2.4.3, "Default Policy Store Permissions for WebCenter Portal"
- Section 30.2.4.4, "Default Code-based Grants"

30.2.4.1 Permission-based Authorization

Permission-based authorization is used for tools, such as lists, where access control is implemented within the WebCenter Portal or Portal Framework application using Oracle Platform Security Services (OPSS). WebCenter Portal provides extensive user and role management tools with which you can create application roles, and define what permissions should be granted to those roles. For information on managing users and roles in WebCenter Portal, see the "Managing Application Roles and Permissions" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

30.2.4.2 Role-mapping Based Authorization

Tools and services that need to access "remote" (back-end) resources require role-mapping based authorization. For example, for discussions, role mapping is required when users of a WebCenter Portal or Portal Framework application (mapping to one or more application roles) must be mapped to another set of roles on the discussions server.

For example, in the WebCenter Portal application:

- WebCenter Portal roles are mapped to corresponding roles on the back-end discussions server.
- When a user is granted a new WebCenter Portal role, a similar grant (privilege) is granted in the back-end discussions server. For example, when user Pat is granted `Discussions-Create/Edit/Delete` permissions in WebCenter Portal, Pat is granted corresponding permissions in the back-end discussions server.

See also the "Understanding Discussions Server Role and Permission Mapping" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

30.2.4.3 Default Policy Store Permissions for WebCenter Portal

Out-of-the box, WebCenter Portal provides the following default roles:

Default application roles:

- Administrator
- Application Specialist
- Authenticated-User
- Public-User

For more information about the default application roles, see

Default roles in a portal:

- Moderator
- Participant
- Viewer

For more information about the default role within a portal, see Section 43.4.2.1, "Understanding Application Roles."

30.2.4.4 Default Code-based Grants

WebCenter Portal and Portal Framework applications make internal calls to APIs on the security platform that are secured with permission checks. Consequently, the application must be granted appropriate permissions to invoke the OPSS APIs (for example, the permission to access the policy store and grant or revoke permissions (`PolicyStoreAccessPermission`, or grant basic permissions to application roles). For WebCenter Portal, basic application role permissions are granted by default as described in Section 43.4.2, "Understanding Application Roles and Permissions."

Similarly, WebCenter Portal and Portal Framework applications must pre-authorize access to various operations that it wants to expose using the WebCenter Portal permissions, and then invoke the OPSS APIs as privileged actions.

30.2.5 Post-deployment Security Configuration Tasks

After deploying WebCenter Portal or Portal Framework application, you should consider the following security-related configuration tasks for your site:

- **Reassociating the identity store to use an external LDAP**

  By default, WebCenter Portal and Portal Framework applications use an embedded LDAP for their identity store. Although secure, the out-of-the-box embedded LDAP may not scale appropriately for large enterprise production environments. For instructions on how to configure the identity store to use an

external LDAP such as Oracle Internet Directory (OID), see Chapter 31, "Configuring the Identity Store."

Note: By default, WebCenter Portal's discussions server is configured to use the embedded LDAP identity store. All users in the embedded LDAP store can log on to the discussions server, and all users in the `Administrators` group have administrative privileges on the discussions server.

If you reassociate the identity store with an external LDAP server, you must either move the system administrator account to the external LDAP (as described in Section 31.4, "Moving the Administrator Account to an External LDAP Server"), or if you choose not to move the administrator account, you must perform some additional steps to identify the new administrator account for the discussions server as described in Section 31.4.1, "Migrating the Discussions Server to Use an External LDAP."

For WebCenter Portal, both the WebCenter Portal application and Content Server must share the same LDAP server. For more information, see Section 31.5, "Configuring the Oracle Content Server to Share the WebCenter Portal Identity Store LDAP Server."

- **Reassociating the policy store to use an external LDAP or database**

  By default, Portal Framework applications use a file-based `system-jazn-data.xml` policy store to store policy grants. You should consider using an LDAP-based or database policy store. For information on how to configure the policy store to use an LDAP server or database, see Chapter 32, "Configuring the Policy and Credential Store."

- **Configuring WS-Security**

  Although the use of WS-Security adds complexity to the configuration and management of a Portal Framework application and the set of producers it consumes, it helps ensure the security of the information being published by the Portal Framework application. Adding WS-Security provides authentication for the consumer, and message-level security.

  For information on how to configure WS-Security for Portal Framework applications and components, see Chapter 36, "Configuring WS-Security."

- **Configuring SSO**

  Single Sign-On (SSO) lets users log in once across WebCenter Portal and Portal Framework applications and components rather than having to log in for each sub-application (for example, for accessing a wiki page in WebCenter Portal). Users do not have to maintain a separate user ID and password for each application or component that they access. However, you can still configure a variety of authentication methods, so that more sensitive applications can be protected using more stringent methods. WebCenter Portal and Portal Framework applications support four single sign-on solutions: Oracle Access Manager (OAM), Oracle Single Sign-on (OSSO), a SAML-based single sign-on solution for Oracle WebCenter Portal applications only, and an SSO solution for Microsoft clients, using Windows authentication based on the Simple and Protected Negotiate (SPNEGO) mechanism and the Kerberos protocol. For a discussion of these solutions and an overview of single sign-on, see Chapter 33, "Configuring Single Sign-on."

- **Configuring SSL**

  Secure Sockets Layer (SSL) provides additional security for connections between WebCenter Portal and Portal Framework applications or components by providing an additional authentication layer, and by encrypting the data exchanged. For connections between applications or components where the data exchanged is sensitive, consider securing the connection with SSL. For a list of the connections that can and should be protected with SSL in a production environment, see Chapter 35, "Configuring SSL."

  Note: Using SSL is computationally intensive and adds overhead to a connection. SSL should therefore not be used where it is not required, and is best reserved for production environments.

30.3 Troubleshooting Security Configuration Issues

This section includes the following sub-sections:

30.3.1 WebCenter Portal Application Does Not Find Users in LDAP Provider

Problem

Weblogic Server was configured with an external LDAP provider. Users in the external LDAP can log in to WebCenter Portal, but when you try to assign the administrator role in WebCenter Portal to a user from the external LDAP, no users are found.

Solution

Change the Control Flag for the `DefaultAuthenticator` Authentication Provider to `Sufficient` as described in Chapter 31, "Configuring the Identity Store." Restart the Administration Server and Managed Servers for the domain.

30.3.2 Portal Created with Errors When Logged in as OID User

Problem

When logged in to WebCenter Portal as an OID user (for example, `orcladmin`), and you try to create a portal, the portal gets created but with errors. The error message appears as "`No matching users were found with search string <login user>`".

Solution

The following property is missing in the `jps-config.xml` file:

```
<property name="jps.user.principal.class.name"
value="weblogic.security.principal.WLSUserImpl"/>
```

To fix this:

1. Edit `DOMAIN_HOME/config/fmwconfig/jps-config.xml`.
2. Add this line in the general properties:

   ```
   <property name="jps.user.principal.class.name"
   value="weblogic.security.principal.WLSUserImpl"/>
   ```

3. Restart the `WC_Spaces` server.

30.3.3 Users Cannot Self-Register when WebCenter Portal Configured with Active Directory

Problem

Users cannot self-register with Active Directory after configuring WebCenter Portal to use AD authenticator. When a user tries to self-register, the following error message appears:

"`User not created. Either the user name or the password does not adhere to the registration policy or the identity store is unavailable. Specify the required user credentials or contact your administrator for assistance.`"

Solution

To fix the problem:

1. Set the user name attribute to `sAMAccountName` while configuring Active Directory in the WebLogic Administration Console.
2. Use the HTTPS port of the LDAP and enable the SSL checkbox while configuring Active Directory in the WebLogic Administration Console.

30.3.4 User Made Administrator Does Not Have Administrator Privileges

Problem

After logging in as `orcladmin` and making a user an administrator, after logging out and logging in as that user, the Administrator link is still not available.

Solution

The problem is due to duplicate `cn` entries in the identity store. Since `cn` is mapped to the username attribute, it must be unique. Remove the duplicate from the identity store and the user should have the appropriate `privileges.cn`.

30.3.5 OmniPortlet Producer Authorization Exception in SSO Environment

Problem

OmniPortlet producer receives an authorization exception when it tries to store connection information in the Credential Store Framework (CSF) wallet when WebCenter Portal is configured with SSO.

Solution

Grant the required permissions to `ssofilter.jar` by connecting to the Oracle WebCenter Portal Administration Server using WLST (for more information, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands") and running the following grant commands:

```
grantPermission(codeBaseURL="file:${oracle.home}/modules/oracle.ssofilter_11.1.1/s
sofilter.jar",
permClass="oracle.security.jps.service.credstore.CredentialAccessPermission",
permTarget="context=SYSTEM,mapName=omniportlet_user,keyName=*",
permActions="*")

grantPermission(codeBaseURL="file:${oracle.home}/modules/oracle.ssofilter_11.1.1/s
sofilter.jar",
permClass="oracle.security.jps.service.credstore.CredentialAccessPermission",
permTarget="context=SYSTEM,mapName=omniportlet_default,keyName=*",
permActions="*")
grantPermission(codeBaseURL="file:${oracle.home}/modules/oracle.ssofilter_11.1
.1/ssofilter.jar",
permClass="oracle.security.jps.service.credstore.CredentialAccessPermission",
permTarget="context=SYSTEM,mapName=omniportlet_user,keyName=*",
permActions="*")
```

30.3.6 Deploying the SAML SSO-specific Discussions EAR file Produces an Exception

Problem

Undeploying the discussions EAR file and deploying the SAML SSO-specific discussions EAR file and then starting the application in the WLS Administration Console produces the following exception:

```
java.lang.ClassCastException:
org.apache.xerces.parsers.XIncludeAwareParserConfiguration
```

Solution

Restart the `WC_Collaboration` server. This should fix the issue and the discussions application will be in an active state.

30.3.7 Configuring SAML Single Sign-on Produces 403 Error

Problem

While testing a SAML SSO configuration you encounter 403 errors, and after turning on debug logging, as described in Section 33.4.2.4, "Checking Your Configuration," you see the following kind of error logs in the destination server:

```
####<Oct 11, 2010 10:20:31 PM PDT> <Debug> <SecuritySAMLLib> <adc2170966>
<soa_server1> <[ACTIVE] ExecuteThread: '1' for queue:
'weblogic.kernel.Default (self-tuning)'> <<WLS Kernel>> <>
```

```
<efaf471a17d5a745:-5ba0524a:12b9b0b7849:-8000-0000000000015385>
<1286860831335> <BEA-000000> <SAMLSignedObject.verify(): validating
signature>
####<Oct 11, 2010 10:20:31 PM PDT> <Debug> <SecuritySAMLService> <adc2170966>
<soa_server1> <[ACTIVE] ExecuteThread: '1' for queue:
'weblogic.kernel.Default (self-tuning)'> <<WLS Kernel>> <>
<efaf471a17d5a745:-5ba0524a:12b9b0b7849:-8000-0000000000015385>
<1286860831336> <BEA-000000> <SAMLDestinationSiteHelper: Signature
verification failed with exception: org.opensaml.InvalidCryptoException:
SAMLSignedObject.verify() failed to validate signature value>
####<Oct 11, 2010 10:20:31 PM PDT> <Debug> <SecuritySAMLService> <adc2170966>
<soa_server1> <[ACTIVE] ExecuteThread: '1' for queue:
'weblogic.kernel.Default (self-tuning)'> <<WLS Kernel>> <>
<efaf471a17d5a745:-5ba0524a:12b9b0b7849:-8000-0000000000015385>
<1286860831336> <BEA-000000> <SAMLDestinationSiteHelper: Unable to validate
response -- returning SC_FORBIDDEN>
####<Oct 11, 2010 10:20:31 PM PDT> <Debug> <SecuritySAMLService> <adc2170966>
<soa_server1> <[ACTIVE] ExecuteThread: '1' for queue:
'weblogic.kernel.Default (self-tuning)'> <<WLS Kernel>> <>
<efaf471a17d5a745:-5ba0524a:12b9b0b7849:-8000-0000000000015385>
<1286860831336> <BEA-000000> <SAMLSingleSignOnService.doACSGet: Failed to get
SAML credentials -- returning>
```

Solution

Chances are that something went wrong with your certificate setup due to which SAML assertions are not being validated. This is likely because the certificate registered in the SAML Identity asserter is incorrect. Export the certificate used for SAML SSO setup in the WebCenter Portal domain specified by `certAlias` and `certPassword` and copy it to a accessible location in the destination domain.

1. Update the relevant config section in the `wcsamlsso.properties` file in the WebCenter Portal domain (for example, if the certificate was invalid for the SOA configuration, update the `certPath` in the `soa_config` section).
2. Open the WebLogic Server Admin Console, and from the `WC_Spaces` domain go to **Security Realm > Providers > Credential Mapping > wcsamlcm > Management > Relying Parties** and delete the relying parties relevant to the domain (for example, for SOA, they would be Worklist Integration, Worklist Detail, and Worklist SDP.)
3. Go to **Destination Domain > Security Realm > Providers > Authentication >wcsamlia > Management > Asserting Parties** and delete the corresponding asserting parties.
4. Open the Certificates tab and delete the certificate as well.
5. Go back to the WebCenter Portal domain and re-run the scripts for creating asserting-relying parties pairs. For SOA, for example, you would need to re-run:

   WCP_ORACLE_HOME/webcenter/scripts/samlsso/configureWorklistIntegration.py
   WCP_ORACLE_HOME/webcenter/scripts/samlsso/configureWorklistDetail.py
   WCP_ORACLE_HOME/webcenter/scripts/samlsso/configureWorklistSDP.py

6. Test your configuration again. If all works well, you can disable SAML logging.

31

Configuring the Identity Store

This chapter describes how to reassociate the identity store with an external LDAP instead of the default embedded LDAP identity store. It also describes how to configure an LDAP server for Oracle WebCenter Content Server and contains the following subsections:

- Section 31.1, "Reassociating the Identity Store with an External LDAP Server"
- Section 31.2, "Configuring the GUID Attribute for External LDAP Identity Stores"
- Section 31.3, "Adding Users to the Embedded LDAP Identity Store"
- Section 31.4, "Moving the Administrator Account to an External LDAP Server"
- Section 31.5, "Configuring the Oracle Content Server to Share the WebCenter Portal Identity Store LDAP Server"
- Section 31.6, "Aggregating Multiple Identity Store LDAP Servers Using libOVD"
- Section 31.7, "Configuring Dynamic Roles for WebCenter Portal"
- Section 31.8, "Configuring Dynamic Groups for WebCenter Portal"
- Section 31.9, "Configuring the REST Service Identity Asserter"

Caution: Before reassociating the identity store, be sure to back up the relevant configuration files:

- `config.xml`
- `jps-config.xml`

As a precaution, you should also back up the `boot.properties` file for the Administration Server for the domain.

Note that for Portal Framework applications, the steps for Migrating the Discussions Server to Use an External LDAP are not required. For more information about the identity store, see the *Oracle Fusion Middleware Application Security Guide*.

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console. Users with the `Monitor` or `Operator` roles can view security information but cannot make changes.

See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

31.1 Reassociating the Identity Store with an External LDAP Server

In almost all cases, you should reassociate the identity store with an external LDAP server rather than using the default embedded LDAP. Although you can use many different types of LDAP servers (see the "Supported LDAP Identity Store Types" section in the *Oracle Fusion Middleware Application Security Guide* for a list of supported LDAPs), this section focuses on how to configure the identity store to use Oracle Internet Directory (OID).

Note: Reassociating the identity store with an external LDAP server is mandatory only if you're using the documents or discussions tools, in which case the `WC_Spaces` server, Content Server, and Collaboration server must all be configured to use the same external LDAP server.

For the GUID attribute for other supported LDAPs, see Section 31.2, "Configuring the GUID Attribute for External LDAP Identity Stores." For other user attribute mappings for supported LDAP servers, see the "The User and Role API" section in the *Oracle Fusion Middleware Application Security Guide*. For information about tuning and performance for LDAPs supported by WebCenter Portal, see the "Tuning Identity Store Configuration" section in the *Oracle Fusion Middleware Performance and Tuning Guide*.

Note: For Oracle databases, Oracle WebCenter Portal only supports the CustomDBMSAuthenticator and ReadOnlySQLAuthenticator authenticators. To use the SQL Authenticator requires a custom SQL Provider based on the User and Role API.

Caution: Reassociating an external LDAP identity store (such as OID) in a production environment with another external LDAP store is not supported. If you have a business need to carry out such a reassociation, please contact Oracle support before going ahead as user information and artifacts may be lost in the process.

To reassociate the identity store with OID:

1. Log in to the WebLogic Server Administration Console.

   For information on logging into the WebLogic Server Administration Console, see Section 1.13.2, "Oracle WebLogic Server Administration Console."

2. In the Domain Structure pane click **Security Realms**.

   The Summary of Security Realms pane displays.

3. In the Name column, click the realm for which you want to reassociate the identity store.

   The Realm Settings pane displays.

4. Open the **Providers** tab.

   The Providers Settings pane displays.

5. Click **New** to add a new provider.

   The Create a New Authentication Provider pane displays.

6. Enter a name for the provider (for example `OIDAuthenticator` for a provider that authenticates the user for the Oracle Internet Directory).
7. Select the authenticator appropriate for your LDAP directory from the list of authenticators.

   Be sure to select the authenticator associated with the LDAP you are configuring rather than choosing the generic `DefaultAuthenticator`. For example, for OID select `OracleInternetDirectoryAuthenticator`, or for iPlanet select `IPlanetAuthenticator`.
8. Click **OK** to save your settings.

   The Settings pane displays with the new authentication provider.
9. In the list of Authentication Providers, click the newly created provider.

   The Settings Pane for the new authentication provider displays.
10. Set the Control Flag to `SUFFICIENT`.

    Setting the Control Flag to `SUFFICIENT` indicates that if a user can be authenticated successfully by this authenticator, then the authentication provider should accept that authentication and should not invoke any additional authenticators.

    Note: If the authentication fails, it falls through to the next authenticator in the chain. Therefore, be sure all subsequent authenticators also have their control flag set to `SUFFICIENT`.
11. Click **Save** to save this setting.
12. Open the Provider Specific tab to enter the details for the LDAP server.

    The Provider Specific pane displays.
13. Enter the details specific to *your* LDAP server.

    Note: The table below shows values appropriate for OID. For the permissible values for other LDAPs, such as Active Directory, see the "OPSS System and Configuration Properties" appendix in the *Oracle Fusion Middleware Application Security Guide*.

Parameter	Value	Description
Host:		The LDAP server's server ID (for example, <ldap_host>`example.com`)
Port:		The LDAP server's port number (for example, `3060`)
Principal:		The LDAP user DN used to connect to the LDAP server (for example, cn=orcladmin)
Credential:		The password used to connect to the LDAP server
User Base DN:		Specify the DN under which your Users start (for example, `cn=users,dc=example,dc=com`)

Parameter	Value	Description
Group Base DN:		Specify the DN that points to your Groups node (for example, `cn=groups,dc=example,dc=com`)
Use Retrieved User Name as Principal	Checked	Must be turned on
All Users Filter:	`(&(uid=*)(objectclass=person))`	Search to find all users under the **User Base DN**
User From Name Filter:	`(&(uid=%u)(objectclass=person))`	
User Name Attribute:	uid	

14. Click **Save**.
15. Return to the Providers tab and reorder the providers so that the new authentication provider is on top, followed by any other authenticators with the `DefaultAuthenticator` placed at the end of the list.

    All should have their control flags set to `SUFFICIENT` so that subsequent authenticators can authenticate identities that fall through from the new provider all the way through to the `DefaultAuthenticator` (which is used only for the default file-based embedded LDAP). For example, logins such as the default administrator account are not typically created in the LDAP directory, but still need to be authenticated to start up the server. Unless identities are allowed to fall through to the `DefaultAuthenticator`, the default administrator account will not be authenticated. For more information about the `DefaultAuthenticator` and the default administrator account, see Section 31.4, "Moving the Administrator Account to an External LDAP Server."

    Note: Do not use the REQUIRED control flag if you are using multiple authenticators. If a REQUIRED control flag is found in the list of authenticators, regardless of its position, no further authenticators will be examined.

16. Restart the Administration Server and the managed server for the changes to take effect.

31.2 Configuring the GUID Attribute for External LDAP Identity Stores

This section describes the different GUID attributes used by non-Oracle LDAP implementations. For other user attribute mappings for other supported LDAP servers, see the "Mapping User Attributes to LDAP Directories" section in the *Oracle Fusion Middleware Application Security Guide*. See also the "Mapping User Attributes to LDAP Directories" section in the *Oracle Fusion Middleware Application Security Guide*.

Note: If you are using an LDAP identity store that does not use the `orclGuid` attribute, such as IBM Tivoli, you can map the `GUID` attribute in the WLS authenticator and it will be used automatically.

IBM Tivoli® Directory Server:

`ibm-entryUUID`

Microsoft® Active Directory:

`objectGUID`

If you are using Active Directory, remember that the `samAccountName` attribute has a 20-character limit; other IDs used by Lotus Connections have a 256-character limit.

Microsoft Active Directory Application Mode (ADAM):

`objectGUID`

To use `objectSID` as the default for ADAM, add the following line to the `<config:attributeConfiguration>` section of the `wimconfig.xml` file:

```
<config:externalIdAttributes name="objectSID" syntax="octetString"/>
```

BM Domino® Enterprise Server:

`dominoUNID`

Note that if the bind ID for the Domino LDAP does not have sufficient manager access to the Domino directory the Virtual Member Manager (VMM) does not return the correct attribute type for the Domino schema query; DN is returned as the VMM ID. To override VMM's default ID setting, add the following line to the `<config:attributeConfiguration>` section of the `wimconfig.xml` file:

```
<config:externalIdAttributes name="dominoUNID"/>
```

Sun Java™ System Directory Server:

`nsuniqueid`

eNovell Directory Server:

`GUID`

31.3 Adding Users to the Embedded LDAP Identity Store

For development or testing purposes, you can add users to the embedded LDAP using the WebLogic Server Administration Console, or using an LDIF file and LDAP commands. Using an LDIF file lets you add additional attributes not available through the WebLogic Server Administration Console.

> **Note:** The embedded LDAP server should only be used for testing or "proof of concept." For production use, Oracle recommends using external identity stores, such as Oracle Internet Directory or Microsoft Active Directory, that are supported by the OPSS user and role APIs. For information about the user and role attributes, see the "Mapping User Attributes to LDAP Directories" section in the *Oracle Fusion Middleware Application Security Guide*.

For Oracle Internet Directory, users are typically managed using ODSM (described in the "Managing Directory Entries" section in *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory*).

Note: If you are planning to reassociate your identity store with an external LDAP, perform that step first (as described in Section 31.1, "Reassociating the Identity Store with an External LDAP Server") as when you reassociate the embedded LDAP with OID or other external LDAP implementation users and user artifacts may not be carried forward. Consequently, do not add users to the embedded LDAP with the expectation of moving them to a production environment. The embedded LDAP is intended to be used only as a test environment, and is not intended as a staging environment that can be moved to production.

WebCenter Portal supports self-registration. New users who self-register with WebCenter Portal are added directly to the identity store. For more information about self-registration, see Section 48.11, "Enabling Self-Registration."

Note: Adding users to the identity store is typically a system administrator task and may not be a task for which application-level administrators have the required permissions.

This section includes the following subsections:

- Section 31.3.1, "Adding Users to the Identity Store Using the WLS Administration Console"
- Section 31.3.2, "Adding Users to the Identity Store Using an LDIF File"

31.3.1 Adding Users to the Identity Store Using the WLS Administration Console

To add users to the embedded LDAP identity store from the WebLogic Server Administration Console:

1. Log in to the WebLogic Server Administration Console.

   For information on logging into the WebLogic Server Administration Console, see Section 1.13.2, "Oracle WebLogic Server Administration Console."

2. In the Domain Structure pane, click **Security Realms**.

   The Summary of Security Realms pane displays.

3. In the Name column, click the realm to which you want to add users.

   The Realm Settings pane displays.

4. Click the **Users and Groups** tab to display the list of current users.

5. Click **New** to add a new user.

6. On the Create a New User page, enter the new user login name in the **Name** field.

   User names are case sensitive and must be unique. Do not use commas, tabs or any of the other characters in the following comma-separated list:

   < >, #, |, &, ?, ( ), { }

7. In the **Description** field, enter a description for the user (for example, the user's full name).

8. From the **Provider** drop-down menu, select `DefaultAuthenticator`.

9. In the **Password** field, enter a password for the user.

   The minimum password length for a user defined in the WebLogic Authentication provider is 8 characters (note that other LDAP providers may have different requirements for the password length). Do not use user name/password combinations such as weblogic/weblogic in a production environment.

10. Reenter the password in the **Confirm Password** field.

11. Click **OK** to save your changes and add the user.

    The user should now appear in the list of users.

31.3.2 Adding Users to the Identity Store Using an LDIF File

You can add users directly to the embedded LDAP identity store using an LDIF file. Using an LDIF file enables you to specify additional user attributes that are not available through the WebLogic Server Administration Console.

As the embedded LDAP server is a conformant LDAP server, you can use LDAP commands to add or modify users. You can also search the directory, which is useful when exporting and importing user accounts.

To add users to the embedded LDAP using an LDIF file you must perform the following tasks:

- Enable External LDAP Access
- Create an LDIF File
- Add the Users

Enable External LDAP Access

When WebLogic Server is installed, the LDAP access credential is set as a randomized value and encrypted in the `config.xml` file. To enable external LDAP access, you must reset the access credential for the embedded LDAP.

To reset the access credential for the embedded LDAP:

1. Log in to the WebLogic Server Administration Console.

   For information on logging into the WebLogic Server Administration Console, see Section 1.13.2, "Oracle WebLogic Server Administration Console."

2. In the Domain Structure pane, click **wc_domain**.

3. In the Settings pane for `wc_domain`, click the Security tab, and then click the Embedded LDAP tab.

   The Settings Pane for `wc_domain` displays the embedded LDAP settings.

4. Enter a new password in the **Credential** field, and reenter it in the **Confirm Credential** field.

5. Click **Save** to save your settings.

6. Restart the WebLogic server.

   After this, you are ready to access the LDAP server with the following values:

   - the DN value for admin access is "cn=Admin"
   - the password is the value you entered in the Credential field
   - the port is the same as the admin port, which by default is 7001

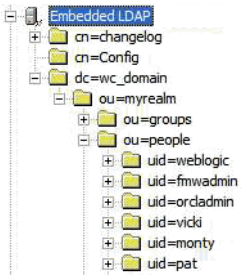
Create an LDIF File

You can create an LDIF file with any text editor, and can include any attributes appropriate for the embedded LDAP directory. The `objectclasses` that are supported by default in the embedded LDAP server for WebLogic Server are the following:

- `person`
- `inetOrgPerson`
- `organizationalPerson`
- `wlsUser`

In order to interact successfully with the embedded LDAP server, you should understand the default layout of the directory information tree (DIT). The default layout in the embedded LDAP directory is shown in Figure 31–1.

Figure 31–1 Embedded LDAP Directory Information Tree



Note: The naming attribute for the user entry in the embedded LDAP directory tree is "uid". This is different from the default configuration for Oracle Internet Directory (OID), where the naming attribute is "cn". Also, the location of the users in this tree is "ou=people,ou=myrealm,dc=wc_domain".

The following example shows an LDIF file with the attributes that are displayed in the WebCenter Portal user profile screens:

```
dn: uid=john.doe,ou=people,ou=myrealm,dc=wc_domain
description: John Doe
cn: john.doe
uid: john.doe
sn: Doe
objectclass: wlsUser
objectclass: organizationalperson
objectclass: inetOrgPerson
objectclass: person
objectclass: top
userpassword: MyPassword
displayName: John Doe
employeeNumber: 12345
employeeType: Regular
givenName: John
homePhone: 650-555-1212
mail: john.doe@example.com
title: Manager
manager: uid=mary.jones,ou=people,ou=myrealm,dc=wc_domain
preferredLanguage: en
```

```
departmentNumber: tools
facsimiletelephonenumber: 650-555-1200
mobile: 650-500-1200
pager: 650-400-1200
telephoneNumber: 650-506-1212
postaladdress: 200 Oracle Parkway
l: Redwood Shores
homepostaladdress: 123 Main St., Anytown 12345
```

To create a file with multiple user entries, just replicate the above lines as many times as required, with a blank line between entries.

Note: WebCenter Portal user profiles include some attributes that are only available in Oracle Internet Directory. These include the following attributes from the `orclUserV2` objectclass:

- `orclTimeZone`
- `orclDateOfBirth`
- `maidenName`

You cannot add these attributes to an embedded LDAP identity store.

Add the Users

The example below uses the `ldappadd` command, a part of the LDAP command line utilities provided with the Oracle Internet Directory server. For more information about using the `ldappadd` command, see the "Oracle Internet Directory Data Management Tools" section in the *Oracle Fusion Middleware Reference for Oracle Identity Management*. For a complete list of user attribute mappings for LDAP servers supported by WebCenter Portal and Portal Framework applications, see the "Mapping User Attributes to LDAP Directories" section in the *Oracle Fusion Middleware Application Security Guide*.

```
ldapadd -h weblogichost.example.com -p 7001 -D cn=Admin -w password -v -f
newuser.ldif

add description:
        John Doe
add cn:
        john.doe
add uid:
        john.doe
add sn:
        Doe
add objectclass:
        wlsUser
        organizationalperson
        inetOrgPerson
        person
        top
add userpassword:
        password
add displayname:
        John Doe
add employeenumber:
        12345
add employeetype:
        Regular
```

```
add givenname:
        John
add homephone:
        650-555-1212
add mail:
        john.doe@example.com
add title:
        Manager
add manager:
        uid=mary.jones,ou=people,ou=myrealm,dc=wc_domain
add preferredlanguage:
        en
add departmentnumber:
        tools
add facsimiletelephonenumber:
        650-555-1200
add mobile:
        650-500-1200
add pager:
        650-400-1200
add telephonenumber:
        650-506-1212
add postaladdress:
        200 Oracle Parkway
add l:
        Redwood Shores
add homepostaladdress:
        123 Main St., Anytown 12345
adding new entry uid=john.doe,ou=people,ou=myrealm,dc=wc_domain
modify complete
```

31.4 Moving the Administrator Account to an External LDAP Server

When configuring the domain to use an external LDAP server, you can also optionally move the system administrator account (`weblogic` by default) to the LDAP server.

If the system administrator account, or any other appropriate user in LDAP, is in an LDAP group called "Administrators", then this account should be sufficient to manage the server, and the `DefaultAuthenticator` provider can be removed from the list of authentication providers. In this case, all users, including the administrator account, are authenticated against the external LDAP.

> **Note:** WebCenter Portal only recognizes users in the identity store that is mapped by the first authenticator. Since the WebCenter Portal Administrator account is initially created only in the embedded LDAP server, if an external LDAP such as Oracle Internet Directory is configured as the primary authenticator for WebCenter Portal, you must also create a user in that LDAP and grant that user the WebCenter Portal Administrator role. For more information about granting the WebCenter Portal Administrator role to a user, see Section 32.6.1, "Granting the WebCenter Portal Administrator Role."

If you cannot create the `weblogic` (default) user in the external LDAP directory, there are two options. You can:

- Keep the `DefaultAuthenticator` provider and use the `weblogic` account with the local embedded LDAP server in WebLogic Server to start and stop

servers and do other administrator operations from the WebLogic Server Administration Console. If you keep the `DefaultAuthenticator`, make sure that the control flag for the `DefaultAuthentication` provider is set to `SUFFICIENT`. If you choose this option, you must also perform the additional steps described in Section 31.4.1, "Migrating the Discussions Server to Use an External LDAP."

Note: If the `weblogic` user account is used from the `DefaultAuthenticator`, this account should not be used to access WebCenter Portal as the application code will not be able to find the user in the external LDAP store.

- Remove the `DefaultAuthenticator` and make sure that any valid user account used for administrator operations, such as starting and stopping servers, is included in an "Administrators" group or other named group that contains the list of users that are allowed to manage your domain in OID or other external LDAP. If a name other than "Administrators" is used, then you must update the group name in the definition of the WebLogic Server Global Administrator role. By default, this is defined as membership in the enterprise group called "Administrators". For information about changing the administrator group name, see Section 31.4.2, "Changing the Administrator Group Name."

This section includes the following subsections:

- Section 31.4.1, "Migrating the Discussions Server to Use an External LDAP"
- Section 31.4.2, "Changing the Administrator Group Name"

31.4.1 Migrating the Discussions Server to Use an External LDAP

If you've installed the discussions server and choose **not to move** the administrator account to an external LDAP (as described in Section 31.4, "Moving the Administrator Account to an External LDAP Server"), you must perform some additional steps to identify the new administrator account for the discussions server prior to reordering the authenticators on the WebLogic server:

1. Select a user account from the external LDAP to be the administrator for the discussions server.
2. Create an administrator account in the `DefaultAuthenticator` (that is, the embedded LDAP) that matches the one you selected from the external LDAP. The account names in the embedded LDAP and the external LDAP server must be the same.

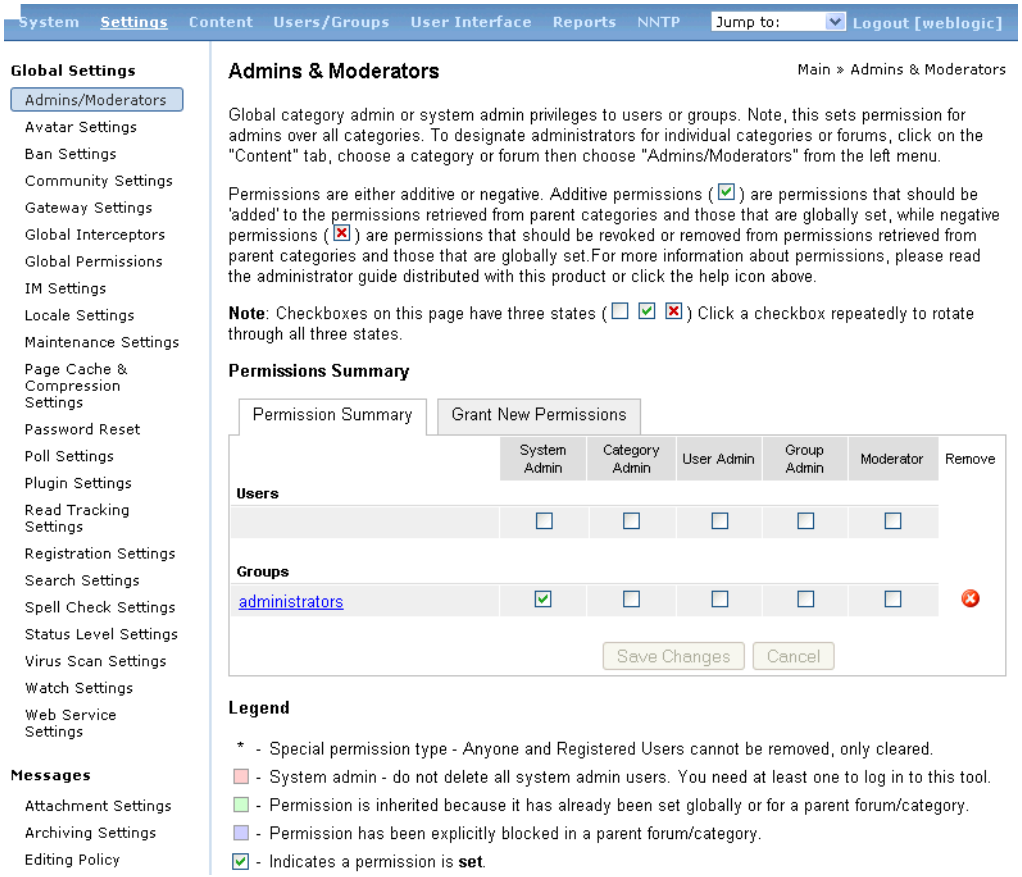
   For information about adding users to the embedded LDAP, see Section 31.3, "Adding Users to the Embedded LDAP Identity Store."
3. Log in to the discussions server Admin Console with the boot-identity account (that is, `weblogic`) at:

   `http://`*`host`*`:`*`port`*`/owc_discussions/admin`

   Where *`host`* and *`port`* are the host ID and port number of the `WLS_Services` managed server.
4. Click **Settings > Admins/Moderators**.

   The Admins & Moderators page displays (see Figure 31–2).

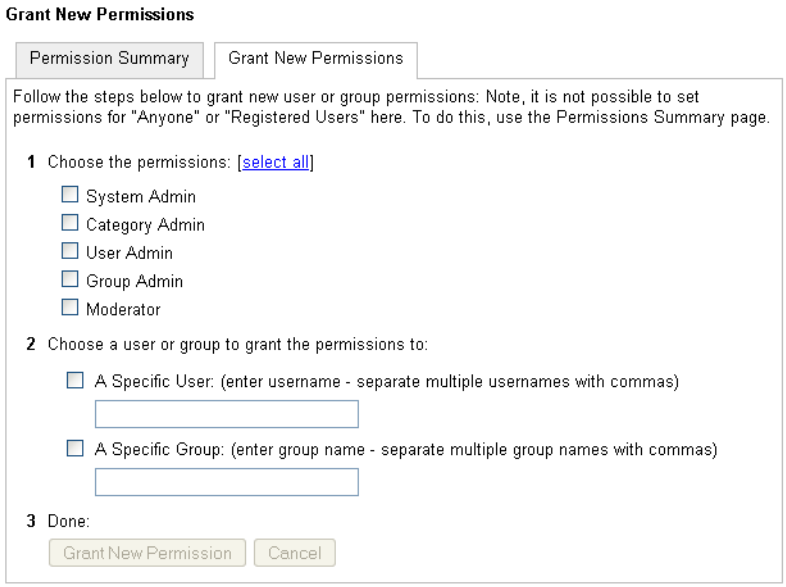
Figure 31–2 Admins & Moderators Page



5. Click **Grant New Permissions**.

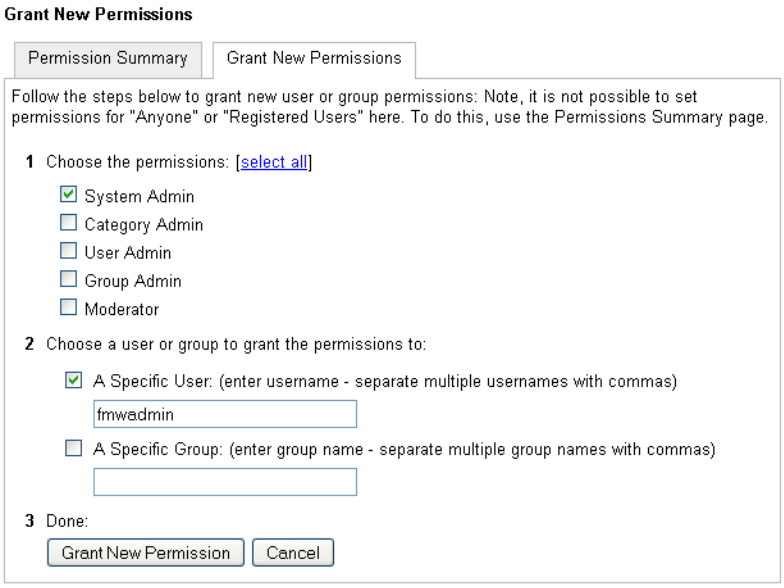
   The Grant New Permissions pane displays (see Figure 31–3).

Figure 31–3 Grant New Permissions Pane



6. Grant System Admin privileges to the user you created, as shown in Figure 31–4.

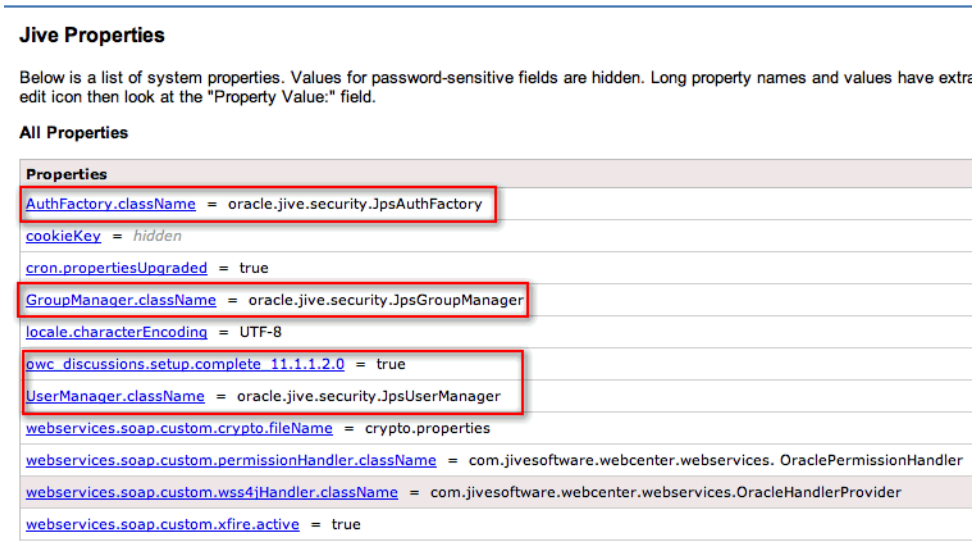
Figure 31–4 Grant New Permissions Pane with New User



7. Click **System > System Properties**.

   The Jive Properties page displays (see Figure 31–5).

Figure 31–5 Jive Properties Page



8. Check that the properties marked in red have been added and are set as shown in Figure 31–5.

9. Log in to the WebLogic Server Administration Console.

   For information on logging in to the WebLogic Server Administration Console, see Section 1.13.2, "Oracle WebLogic Server Administration Console."

10. In the Domain Structure pane, click **Security Realms**.

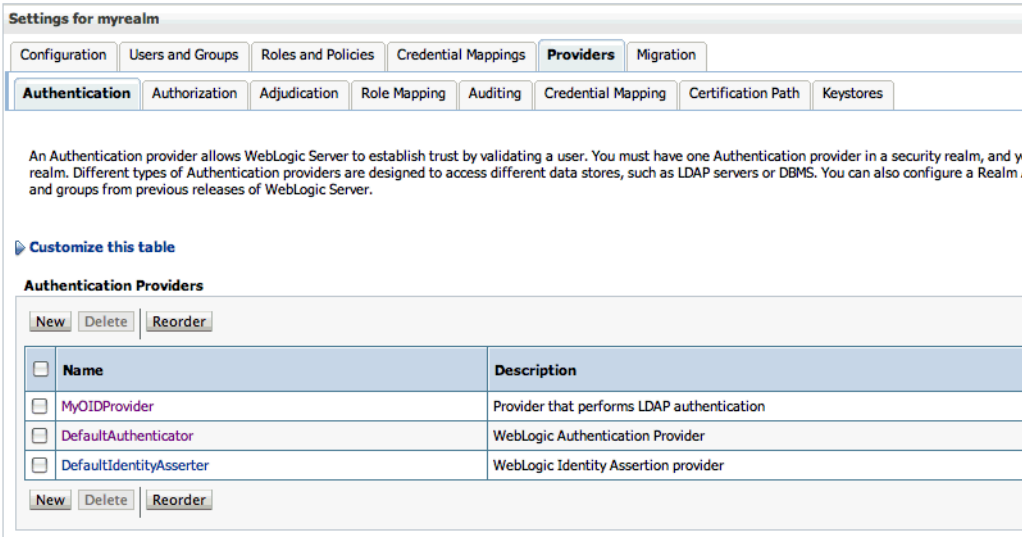
The Summary of Security Realms pane displays.

11. In the Name column, click the realm for which you want to change the administrator group name.

    The Realm Settings pane displays.

12. Select the Providers tab and the Authentication sub-tab, and reorder the authentication providers so that the authenticator for the external LDAP appears at the top of the list as shown in the example in Figure 31–6:

Figure 31–6 Providers Tab with Reordered Authentication Providers



13. Restart the domain Administration server and discussions server.

14. If you have not done so already, create a user in the external LDAP and grant that user the WebCenter Portal Administrator role (see Section 32.6.1, "Granting the WebCenter Portal Administrator Role").

31.4.2 Changing the Administrator Group Name

You can change the group name to any other valid enterprise role in your LDAP server that contains users authorized to manage the domain. This lets you delegate the administration of specific domains in your enterprise. You can create various administration groups in the directory and have the corresponding domains be configured to use the appropriate group for defining its administrators.

The following example LDIF file creates an administrative group in Oracle Internet Directory:

```
dn: cn=wc_domain_Admin,cn=groups,dc=example,dc=com
cn: wc_domain_Admin
uniquemember: cn=joe.admin,cn=users,dc=example,dc=com
owner: cn=orcladmin
displayname: WebLogic Administrators Group
description: WebLogic Administrators Group
objectclass: orclgroup
objectclass: groupofuniquenames
```

Once this group is created, you must update the role definition for the WebLogic Server global Admin role using the WebLogic Server Administration Console.

To update the role definition for the WebLogic Server global Admin role:

1. Log in to the WebLogic Server Administration Console.

   For information on logging into the WebLogic Server Administration Console, see Section 1.13.2, "Oracle WebLogic Server Administration Console."

2. In the Domain Structure pane, click **Security Realms**.

   The Summary of Security Realms pane displays.

3. In the Name column, click the realm for which you want to change the administrator group name.

   The Realm Settings pane displays.

4. Open the Roles and Policies tab, and then the Realm Roles subtab.

   The Realm Roles settings pane displays.

5. Expand the Global Roles node, and then the Roles node.

6. Click **View Role Conditions** for the `Admin` role.

   The Edit Global Role page displays.

   By default, the `Administrators` group in Oracle Internet Directory (or other configured identity store) defines who has the administrator role in WebLogic Server.

7. Click **Add Conditions** to add a different group name.

   The Edit Global Role - Predicate List page displays.

8. Select `Group` from the **Predicate List** list and click **Next**.

   The Edit Global Role - Arguments page displays.

9. Enter the name for the new administrator group and click **Add**.

10. Select the pre-existing administrator group and click **Remove** to delete it leaving the new one you've selected in its place.

11. Click **Finish** to save your changes.

    After making this change, any members of the new group specified are authorized to administer WebLogic Server.

31.5 Configuring the Oracle Content Server to Share the WebCenter Portal Identity Store LDAP Server

Oracle Content Server (OCS) must be configured to use the same identity store LDAP server as WebCenter Portal. For more information on configuring the OCS, see Chapter 9, "Managing Content Repositories" and also see the "Configuring the LDAP Identity Store Service" section in the *Oracle Fusion Middleware Application Security Guide*.

31.6 Aggregating Multiple Identity Store LDAP Servers Using libOVD

Sites with multiple identity stores can use libOVD to aggregate their user profile information. Two scenarios are covered in the step-by-step configuration instructions below:

- Users are available in distinct identity stores with complete user profile information available in the respective identity store.

- The same user is available in both identity stores with some attributes in one store and other attributes in the other store.

> **Note:** If you are supporting self-registration with Active Directory, be sure to see the troubleshooting note in Section 30.3.3, "Users Cannot Self-Register when WebCenter Portal Configured with Active Directory."

This section contains the following subsections:

- Section 31.6.1, "Configuring libOVD for Identity Stores with Complete User Profiles"
- Section 31.6.2, "Configuring libOVD for Identity Stores with Partial User Profiles"
- Section 31.6.3, "Restoring the Single Authenticator"

31.6.1 Configuring libOVD for Identity Stores with Complete User Profiles

To configure libOVD where each identity store contains complete user profiles:

1. Create the required authenticators in the WLS Admin Console for the identity stores being configured and restart the Weblogic Admin and managed servers for the domain. Alternatively, you can also configure the identity store information in `jps-config.xml` by hand.
2. Update the identity store service instance in `jps-config.xml` and add a property `virtualize` with the value `true`. You can do this either by editing the `jps-config.xml` file by hand, or using Fusion Middleware Control.
3. WebCenter Portal lets users self-register, which creates a new user or group in the identity store. Since multiple identity stores are being used, you also need to explicitly specify the user create bases and group create bases in `jps-config.xml`. This step must be done by directly editing `jps-config.xml`.

   The `jps-config.xml` file should look like the example below after the configuration.

```
<serviceInstance provider="idstore.ldap.provider" name="idstore.ldap">
<property
value="oracle.security.jps.wls.internal.idstore.WlsLdapIdStoreConfigProvider"
name="idstore.config.provider"/>
<property value="oracle.security.idm.providers.stdldap.JNDIPool"
name="CONNECTION_POOL_CLASS"/>
<property value="true" name="virtualize"/>
<extendedProperty>
        <name>user.create.bases</name>
        <values>
             <value>ou=people,ou=myrealm,dc=wc_domain</value>
         </values>
</extendedProperty>
<extendedProperty>
        <name>group.create.bases</name>
        <values>
              <value>ou=groups,ou=myrealm,dc=wc_domain</value>
        </values>
</extendedProperty>
</serviceInstance>
```

Be sure to replace the actual values for the user create base in "ou=people,ou=myrealm,dc=wc_domain" and group create base "ou=groups,ou=myrealm,dc=wc_domain."

31.6.2 Configuring libOVD for Identity Stores with Partial User Profiles

To configure libOVD where each identity store contains only partial user profiles:

1. Create the required authenticators in the WLS Admin Console for the identity stores being configured and restart the Weblogic Admin and managed servers for the domain. Alternatively, you can also configure the identity store information in jps-config.xml by hand.
2. Update the identity store service instance in jps-config.xml and add a property virtualize with the value true. You can do this either by editing the jps-config.xml file by hand, or using Fusion Middleware Control.
3. WebCenter Portal lets users self-register, which creates a new user or group in the identity store. Since multiple identity stores are being used, you also need to explicitly specify the user create bases and group create bases in jps-config.xml. This step must be done by directly editing jps-config.xml.

   The jps-config.xml file should look like the example below after the configuration.

```
<serviceInstance provider="idstore.ldap.provider" name="idstore.ldap">
<property
value="oracle.security.jps.wls.internal.idstore.WlsLdapIdStoreConfigProvider"
name="idstore.config.provider"/>
<property value="oracle.security.idm.providers.stdldap.JNDIPool"
name="CONNECTION_POOL_CLASS"/>
<property value="true" name="virtualize"/>

<extendedProperty>
        <name>user.create.bases</name>
        <values>
             <value>ou=people,ou=myrealm,dc=wc_domain</value>
         </values>
</extendedProperty>
<extendedProperty>
        <name>group.create.bases</name>
        <values>
              <value>ou=groups,ou=myrealm,dc=wc_domain</value>
        </values>
</extendedProperty>
</serviceInstance>
```

   In the above example "ou=people,ou=myrealm,dc=wc_domain" and "ou=groups,ou=myrealm,dc=wc_domain" are the user and group create bases respectively. The actual values should be substituted while doing the configuration.

4. Run the following OVD WLST commands to configure the Join Adapter for the identity stores. Go to *MW_HOME*/oracle_common/common/bin and invoke wlst.sh (wlst.cmd in windows) and bring up the WLST prompt. Connect to the Weblogic Administration Server and run the following WLST commands.

```
createJoinAdapter(adapterName="<Join Adapter Name>",  root="<Namespace>",
primaryAdapter="<Primary adapter Name>")

addJoinRule(adapterName="<Join Adapter Name>", secondary="<Secondary Adapter
```

```
Name>", condition="<Join Condition>")
```

If there are more secondary identity stores, then run the `addJoinRule` command for each secondary identity store.

```
modifyLDAPAdapter(adapterName="<AuthenticatorName>", attribute="Visible",
value="Internal")
```

Run the above `modifyLDAPAdapter` command for each identity store that is configured.

Example

Authenticator 1:

In this example, the same user is available in both identity stores with some attributes in one store and some in the other. For this example, AD is the primary store and OID is the secondary store.

Authenticator Name: `AD`

User Base: `cn=users,dc=acme,dc=com`

Authenticator 2:

Authenticator Name: `OID`

User Base: `cn=users,dc=oid,dc=com`

Perform steps 1 - 3 above, specifying the `user.create.bases` and `group.create.bases` corresponding to the primary adapter's namespace.

Perform the following WLST commands:

```
createJoinAdapter(adapterName="JoinAdapter1", root="dc=acme,dc=com",
primaryAdapter="AD")
addJoinRule(adapterName="JoinAdapter1", secondary="OID", condition="uid=cn")
```

"`uid=cn`" is the join condition in the above example, which indicates that if the `uid` value of a user in the secondary identity store (OID) matches with the `cn` value of the user in the primary identity store (AD), then the attributes will be combined.

```
modifyLDAPAdapter(adapterName="OID", attribute="Visible", value="Internal")
modifyLDAPAdapter(adapterName="AD", attribute="Visible", value="Internal")
```

Restart the WebLogic Administration server and managed servers.

31.6.3 Restoring the Single Authenticator

You can restore the single authenticator by removing the Join Adapter rule, thereby backing out the configuration done in Section 31.6.2, "Configuring libOVD for Identity Stores with Partial User Profiles."

To remove the Join Adapter rule, connect to the Weblogic Administration Server and run the following WLST commands:

```
deleteAdapter(adapterName="JoinAdapter1")
modifyLDAPAdapter(adapterName="oid auth", attribute="Visible", value="Yes")
modifyLDAPAdapter(adapterName="AD", attribute="Visible", value="Yes")
```

Restart the WebLogic Administration server and managed servers and make sure that users from both identity stores are able to log in.

31.7 Configuring Dynamic Roles for WebCenter Portal

This section describes how to configure dynamic roles for WebCenter Portal.

This section contains the following subsections:

- Section 31.7.1, "Overview of Configuring Dynamic Roles"
- Section 31.7.2, "Prerequisites to Configuring Dynamic Roles"
- Section 31.7.3, "Installing the OVD Plug-in"
- Section 31.7.4, "Configuring Dynamic Roles"

31.7.1 Overview of Configuring Dynamic Roles

With static roles, the role to membership relationship is static. This relationship is established at provisioning time, and once established, and after a user logs in, the subject is populated with all the roles for which the user is a direct member or indirectly a member based on an enterprise group.

Dynamic roles provide for rule-based role membership. Membership to an application role is provided through a dynamic group. Dynamic group definitions can include constraints for user profile attributes, and date and time that provide a flexible way to provide access to an application. For example, a user could be allowed to access an application only during their shift or during maintenance periods without explicitly having to grant them that access. Note that rules based on session or request attributes are not supported in this release.

Dynamic roles can be defined in Oracle Entitlement Server (OES) 10*g* as a role with constraints. The role defined in OES is added to user's enterprise group principal through an OVD plug-in. When the user logs in the policy rules are evaluated to determine whether the user's subject gets the dynamic group principal. Figure 31–7 shows the login process for a topology configured with OES and the OVD plug-in.

Figure 31–7 Login Process

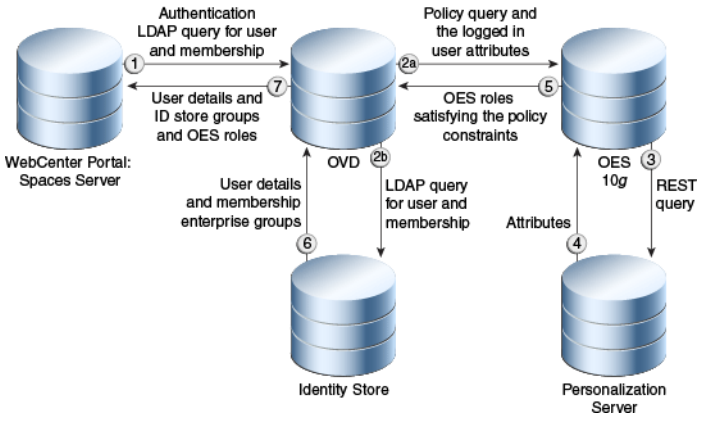
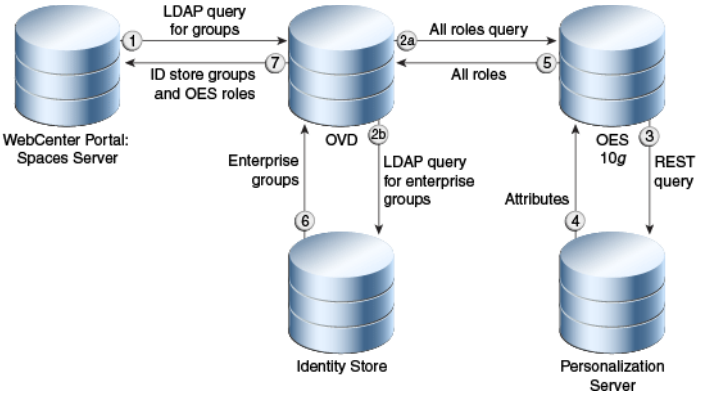


Figure 31–8 shows what happens during a dynamic group query for a topology configured with OES and the OVD plug-in.

Figure 31–8 Group Query



31.7.2 Prerequisites to Configuring Dynamic Roles

Prior to installing and configuring the OVD plug-in, you should already have installed and configured Oracle Internet Directory (OID), Oracle Virtual Directory (OVD) 11*g*, and Oracle Entitlement Server (OES) 10g. Note that the OVD plug-in is not currently certified with OES 11*g*.

OID and OVD are part of the Oracle Identity Management 11*g* suite. If you have not already installed them, install them as described in "Installing and Configuring Oracle Identity Management (11.1.1.7.0)" in *Oracle Fusion Middleware Installation Guide for Oracle Identity Management*. Configure OVD by running `config.sh` as described in the "Configuring Oracle Virtual Directory" section in *Oracle Fusion Middleware Installation Guide for Oracle Identity Management*.

Install OES 10*g* with RMI-SSM and the latest cumulative patch as described in the section "Configuring a Remote SSM and Proxy" in the *SSM Installation and Configuration Guide*.

Additionally, if you have plan to incorporate constraints based on Personalization for WebCenter Portal, you will also need to install and configure the Personalization server, as described in Section 25, "Managing Personalization."

31.7.3 Installing the OVD Plug-in

Oracle Virtual Directory (OVD) is a part of the Oracle Identity Management suite of products. OVD provides an elegant solution to the problem of integrating multiple heterogeneous data sources presenting them as a consolidated view that can be consumed by an LDAP client in WebCenter Portal or Portal Framework applications. Through OVD, OES data can be exposed by means of an OVD custom adaptor in a way that it can be consumed by WebCenter Portal or Portal Framework applications. The adapter can represent non-LDAP data as an LDAP-like tree hierarchy. The functionality of the custom adapter is contained in a plug-in that can be attached to the adapter.

To install the OVD plug-in:

1. Locate the `oes-ovd-plugin.zip` file in the following folder:

   ```
   WCP_HOME/modules/oracle.webcenter.framework_11.1.1/oes-ovd-plugin.zip
   ```

2. Make a copy of the `oes-ovd-plugin.zip` file.

3. Go to the `plugins/lib` directory where OVD was installed (for example, `asinst_1/OVD/ovd1/plugins/lib`).
4. Unzip `oes-ovd-plugin.zip`.
5. Copy `webcenterNames.xml` to the instance home (for example, *`ORACLE_HOME`*`/asinst_1`).
6. Create the following directories:

```
mkdir pdpproxy
mkdir keys
```

7. Copy the following OES jars from the `OES/rmi-ssm` installation directory to the `lib` directory:

```
EccpressoCore.jar
antlr.jar
api.jar
asi_classes.jar
asitools.jar
commons-pool-1.3.jar
connector.jar
framework.jar
jsafeFIPS.jar
jsafeJCEFIPS.jar
kodo-runtime.jar
log4j.jar
managementapi.jar
orawsdl14.jar
rmi-ssm.jar
rmi-stubs.jar
rmi-types.jar
ssladapter.jar
sslplus.jar
webservice.jar
webserviceclient.jar
webservices.jar
xbean.jar
```

8. Go to the `keys` directory that you just created and copy all the keys in the OES install (`ales32-shared/keys` directory) here.
9. Go to the `pdpproxy` directory that you just created and copy the PDP configuration properties file from OES (`rmi-ssm/pdpproxy/PDPProxyConfiguration.properties`).
10. Restart the OVD process with the following command:

```
./opmnctl stopall startall
```

11. If you are planning to use Personalization in defining your constraints, install the `p13nattributeRetriever` as shown below:
    a. Locate the `<WebCenter Home>/webcenter/modules/oracle.webcenter.framework_11.1.1/attribute-retriever.jar` file.
    b. Locate the `rmi-ssm/lib/providers` directory of the OES installation, and copy the `attribute-retriever.jar` file there.
    c. Restart the rmi-ssm.

31.7.4 Configuring Dynamic Roles

This section describes how to configure your WebCenter Portal environment to support dynamic roles through OES and the OVD plug-in. Prior to completing the configuration steps in this section, you should already have installed the prerequisite applications (described in Section 31.7.2, "Prerequisites to Configuring Dynamic Roles") and the OVD plug-in (described in Section 31.7.3, "Installing the OVD Plug-in").

This section contains the following subsections:

- Section 31.7.4.1, "Configuring OES"
- Section 31.7.4.2, "Configuring the OVD Plug-in"
- Section 31.7.4.3, "Configuring the Personalization Attributes"
- Section 31.7.4.4, "Configuring WebCenter Portal to Consume Dynamic Roles"

31.7.4.1 Configuring OES

All the dynamic roles that you want to be available in WebCenter Portal should be defined under a single umbrella resource and action. In the steps below, we're using `WebCenterApp/WebCenterResource` as the umbrella resource, and `browse` as the action. When you create the dynamic roles, the roles are then granted browse permission on the resource using role mapping polices. A role mapping policy can also have additional constraints based on identity store or Personalization attributes.

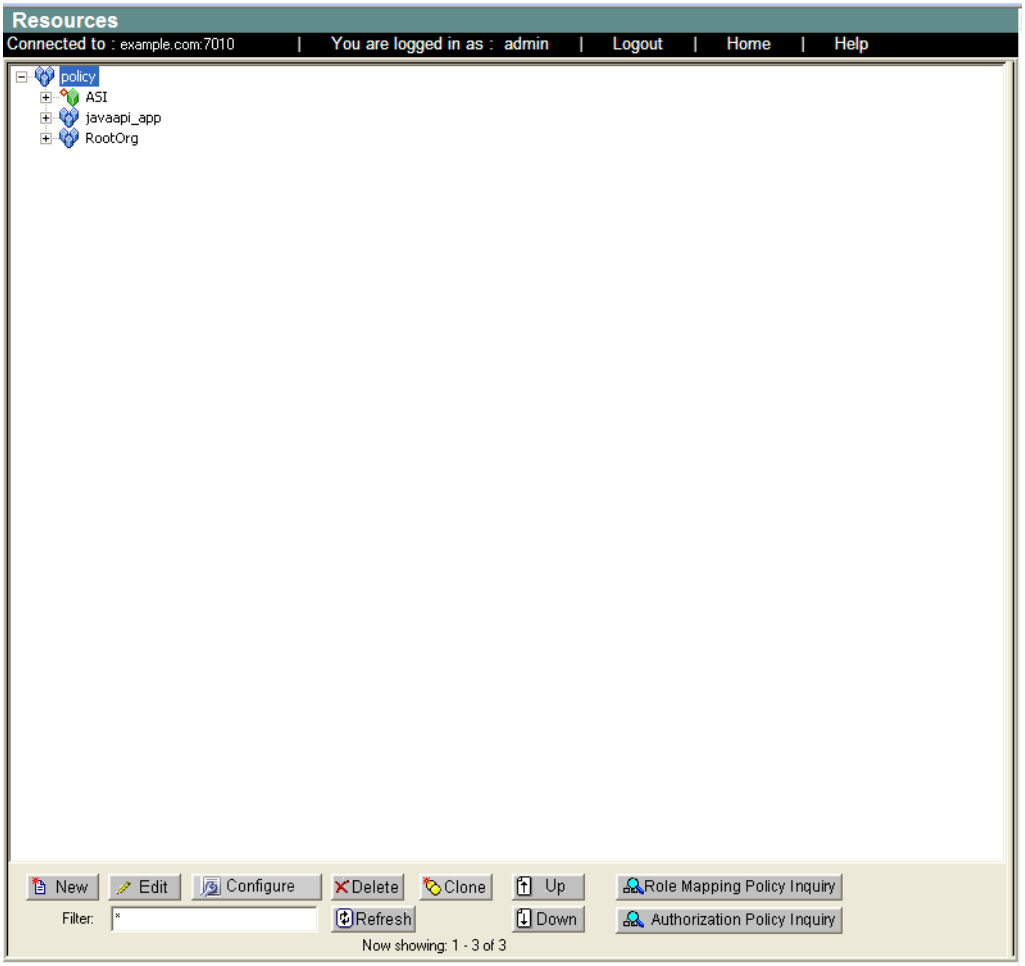
To configure OES for dynamic groups:

1. Open a browser and log onto the OES console as an administrator:

   ```
   https://<host>:<port>/asi
   ```

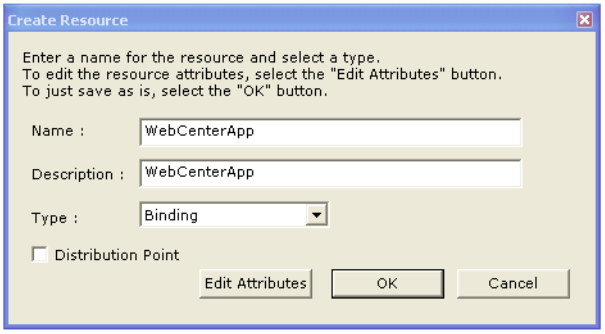
2. Under the Administration Console node, click **Resources** to display a list of the currently defined resources in the Resources page as shown in Figure 31–9.

Figure 31–9 Resources Page



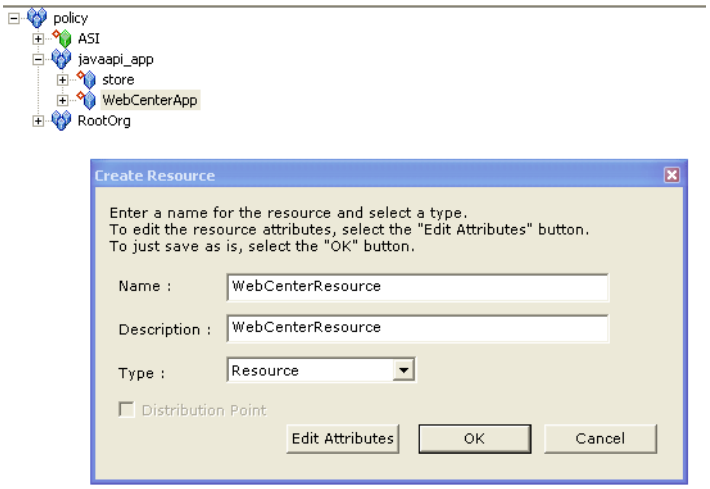
3. Create a resource root by right-clicking the `javaapi_app` node and selecting **Add Resource** to display the Create Resource dialog. Name the resource `WebCenterApp` and select `Binding` as the **Type** as shown in Figure 31–10.

Figure 31–10 Creating a Root Resource



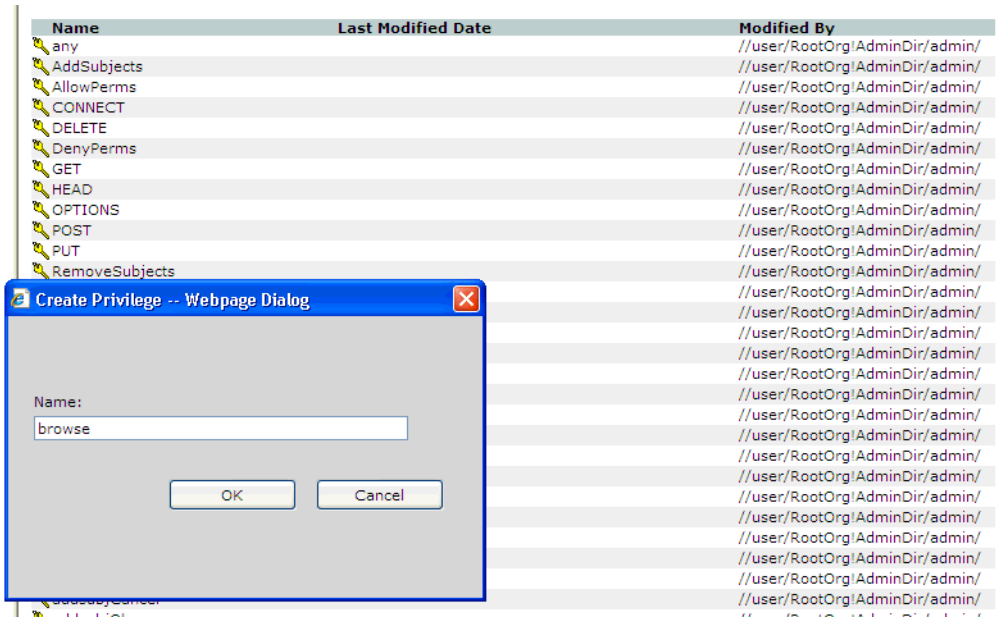
4. Right-click `WebCenterApp` and create a new resource under it naming it `WebCenterResource` and selecting `Resource` as the **Type** as shown in Figure 31–11.

Figure 31–11 Creating a Resource



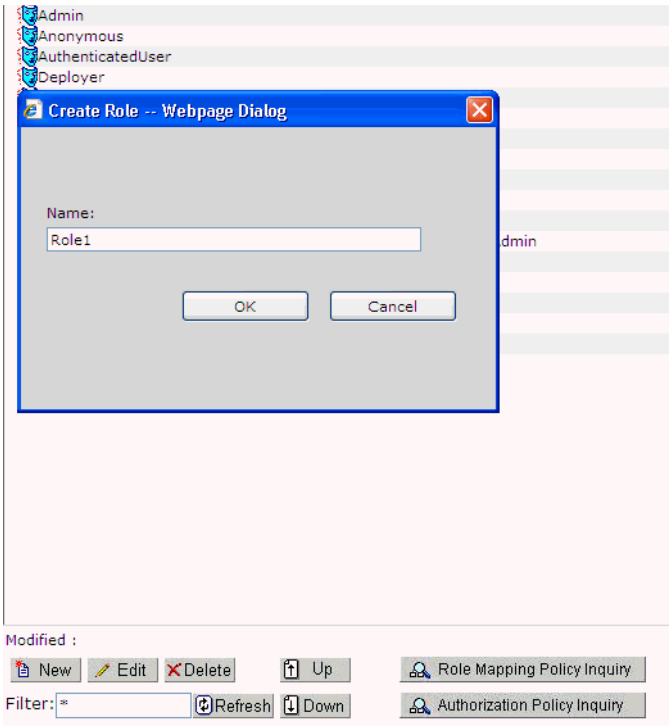
5. Under the Resources node, click **Privileges** to display the Privileges page.
6. Click **New** and create an action naming it `browse` as shown in Figure 31–12.

Figure 31–12 Creating an Action



7. Open the Identity node and click **Roles** to display the Roles page.
8. Click **New** to create the dynamic roles using the Create Role dialog as shown in Figure 31–13.

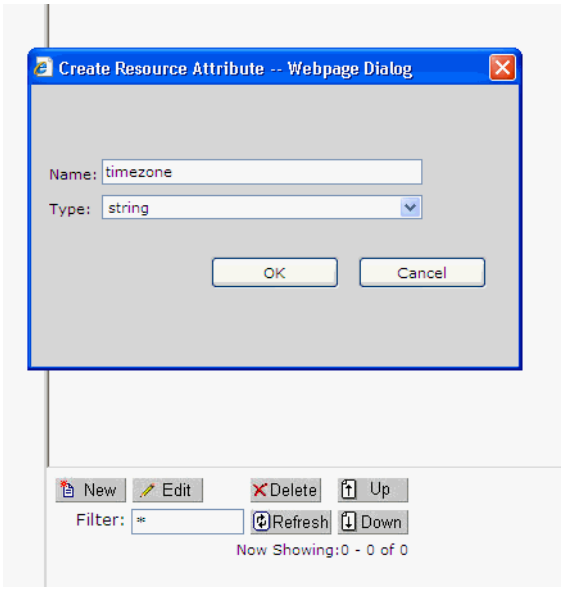
Figure 31–13 Creating the Dynamic Roles



9. Open the Resources node and click **Attributes** to display the Resource Attributes page.

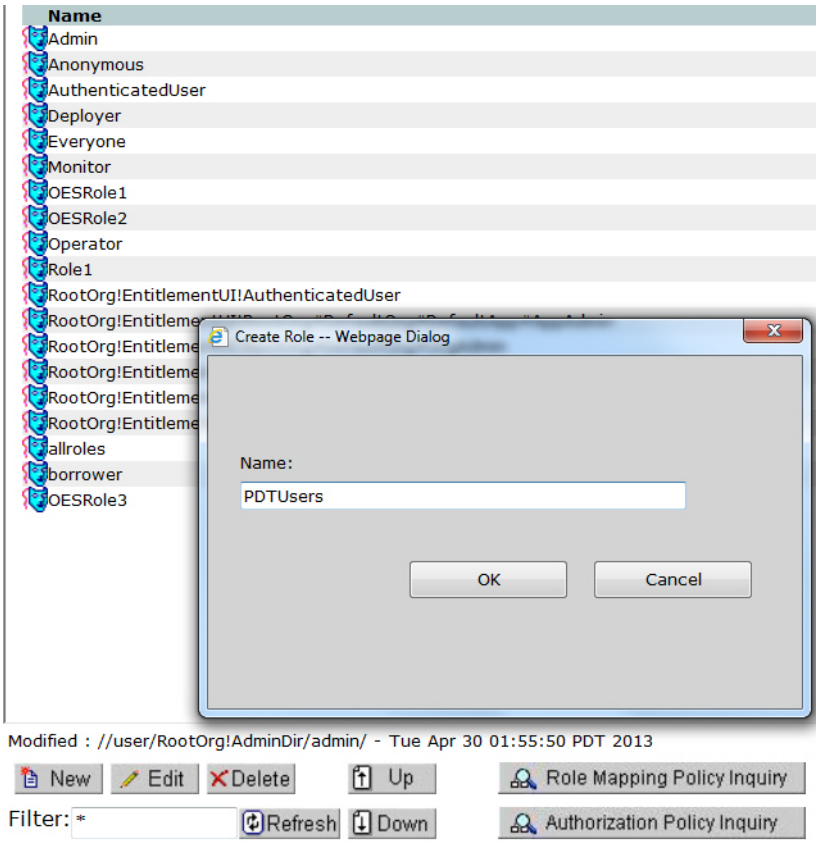
10. Click **New** and use the Create Resource Attribute dialog to create resource attributes using the same name as the identity store attributes that are to be used in the constraints (for example, `business_email` or `timezone`) as shown in Figure 31–14. If Personalization attributes are to be used in the constraints then those attributes should also be created. Note that the attributes that are used as constraints in the Role Mapping policy cannot be empty in the identity store.

Figure 31–14 Creating Resource Attributes



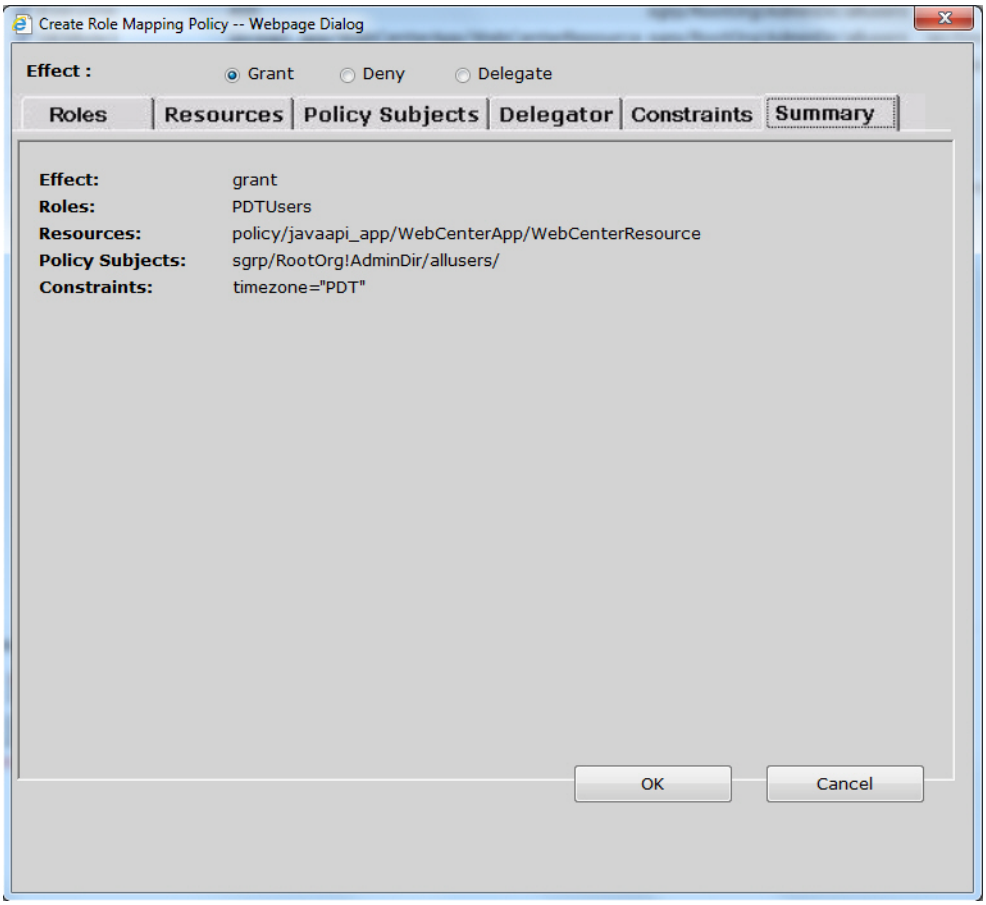
11. Open the Policy node and click **Role Mapping Policies** to display the Role Mapping Policies page.
12. Click **New** to display the Create Role Mapping Policy dialog and create role mapping policies and constraints using the identity store attributes or built-in system attributes (such as `hour` or `day`) as shown in Figure 31–15.

Figure 31–15 Creating Role Mapping Policies



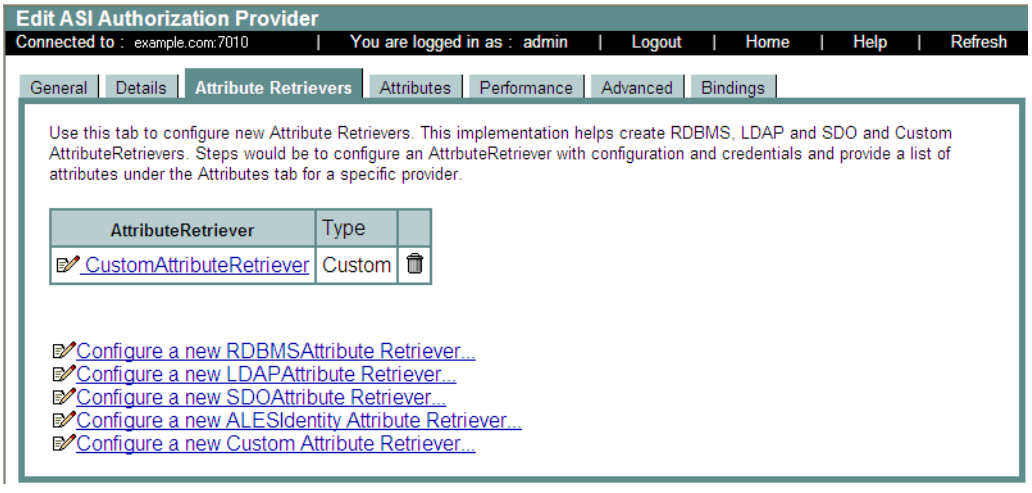
13. Under the Policy node click **Authorization Policies** to display the Authorization Policies page.
14. Click **New** to use the Create Authorization Policies dialog to create new authorization policies, or from the summary of authorization policies select a policy and click **Edit** to edit and provide details for the policy such resources and policy subjects as shown in Figure 31–16.

Figure 31–16 Creating Authorization Policies



15. Open the Authorization node under Service Control Managers, click **ASIAuthorizationProvider** and then open the Attribute Retrievers tab as shown in Figure 31–17.

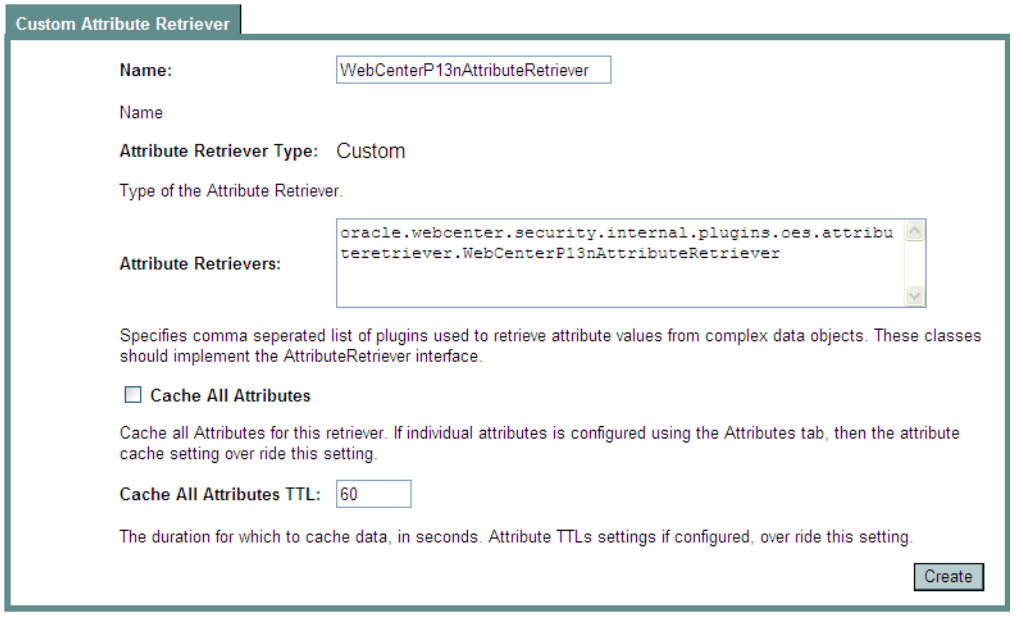
Figure 31–17 Attribute Retrievers Tab



16. Click **Configure a new Custom Attribute Retriever** and create a custom attribute retriever named `WebCenterP13nAttributeRetriever`

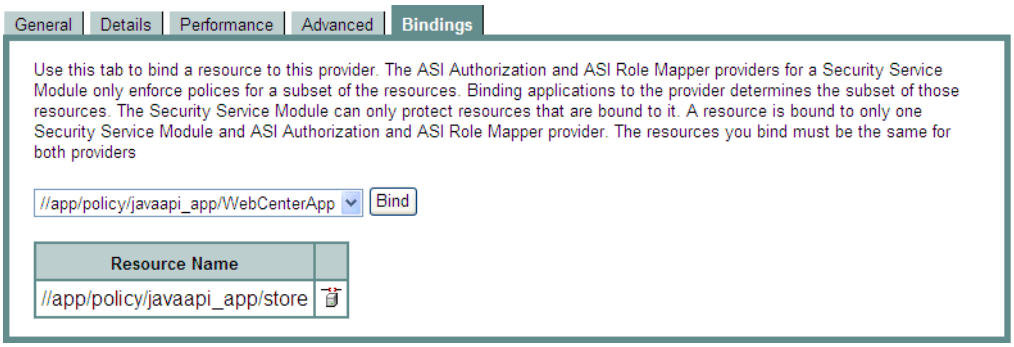
(`oracle.webcenter.security.internal.plugins.oes.attributeretriever.WebCenterP13nAttributeRetriever`), adding the class details as shown in Figure 31–18.

Figure 31–18 Creating a Custom Attribute Retriever



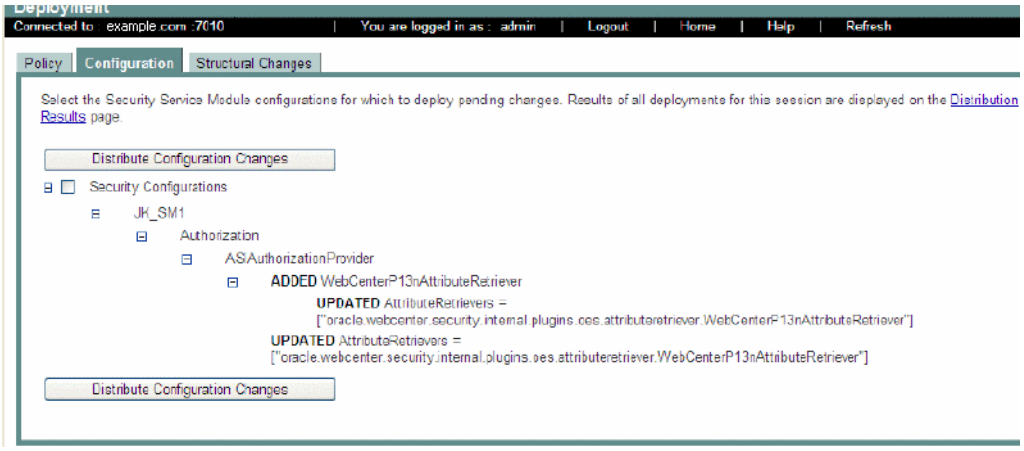
17. Open the Role Mapping node under Service Control Managers, click **ASIRoleMapperProvider** and open the Bindings tab. Bind the WebCenterApp resource to the authorization provider as shown in Figure 31–19.

Figure 31–19 Binding the Resource to the Authorization Provider



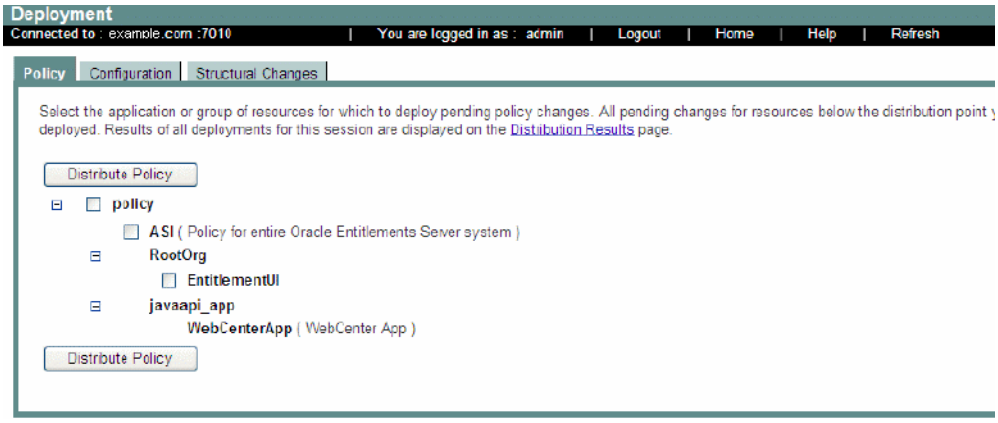
18. Click **Deployment**, open the Configuration tab and distribute the configuration changes as shown in Figure 31–20.

Figure 31–20 Distributing the Configuration Changes



19. Open the Policy tab and distribute the policy changes as shown in Figure 31–21.

Figure 31–21 Distributing the Policy Changes



31.7.4.2 Configuring the OVD Plug-in

This section describes how to configure the OVD plug-in.

To configure the OVD plug-in:

1. Go to the `plugins/lib/pdpproxy` directory and edit the file `PDPProxyConfiguration.properties`, providing the SSM configuration ID, the OES host name, the RMIS port, and the trust store location. Example values are shown below:

```
SSMConfigID=JK_SM1
PDPTransport=RMI
PDPAddress=rmis://example.com:9300 (the use of SSL port is always recommended)
TrustStore=<OID_HOME>/asinst_1/OVD/ovd1/plugins/lib/keys/DemoTrust.jks
```

2. Open the file `./config/OPMN/opmn/opmn.xml` and change the `java-options` and `java-classpath` of the OVD process shown in the following sample, providing the correct OVD home path:

```
<data id="java-options" value="-server -Xms256m -Xmx256m
-Dvde.soTimeoutBackend=0 -Didm.oracle.home=$ORACLE_HOME
-Dcommon.components.home=$ORACLE_HOME/../oracle_common
-Doracle.security.jps.config=$ORACLE_INSTANCE/config/JPS/jps-config-jse.xml
```

```
-Djavax.net.ssl.trustStore=$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/keys/DemoTrust
.jks
-Dpdp.configuration.properties.location=$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/
pdpproxy/PDPProxyConfiguration.properties
-Dwles.ssl.identityKeyAlias=wles-admin
-Dwles.ssl.identityKeyPasswordAlias=wles-admin
-Dwles.ssl.identityKeyStore=$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/keys/identity
.jceks
-Dwles.ssl.trustedCAKeyStore=$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/keys/trust.j
ks
-Dwles.ssl.passwordFile=$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/keys/password.xml
-Dwles.ssl.passwordKeyFile=$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/keys/password.
key"/>

<data id="java-classpath"
value="$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/jsafeFIPS.jar:
$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/jsafeJCEFIPS.jar:
$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/scmapi.jar:
$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/sslplus.jar:
$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/ssladapter.jar:
$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/asitools.jar:
$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/webserviceclient.jar:
$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/EccpressoCore.jar:
$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/webservice.jar:
$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/kodo-runtime.jar:
$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/kodo-runtime.jar:
$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/commons-pool-1.3.jar:
$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/oes-ovd-plugin.jar:
$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/xbean.jar:
$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/antlr.jar:
$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/log4j.jar:
$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/api.jar:
$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/asi_classes.jar:
$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/framework.jar:
$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/rmi-types.jar:
$ORACLE_INSTANCE/OVD/ovd1/plugins/lib/rmi-ssm.jar:
$ORACLE_HOME/ovd/jlib/vde.jar$:$ORACLE_HOME/jdbc/lib/ojdbc6.jar"/>
```

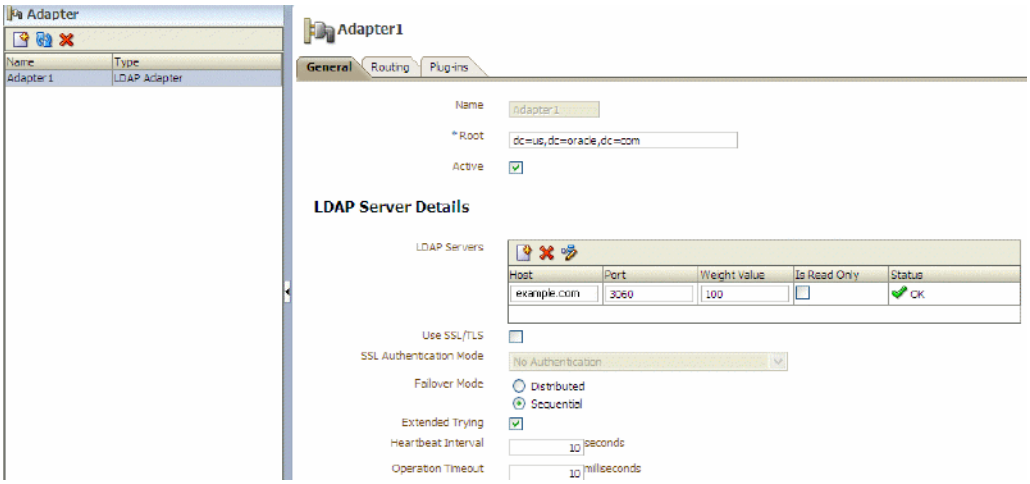
3. Using your browser, open the Oracle Directory Service Manager (ODSM):

   ```
   http://host:port/odsm
   ```

   To determine the ODSM port use the `opmnctl status` command in the OID installation. The default port is `7005`.

4. Create an adapter of **Type** `LDAP Adapter`, providing the LDAP host and port details as shown in the example in Figure 31–22.

Figure 31–22 Example LDAP Adapter



5. Choose the DN and provide a mapping name for the DN.

6. If you need to map to attributes other than the default, open the Plug-ins tab of the adapter and add the UserManagement adapter. For example, if you are using Active Directory as the backend directory server for OVD, add the UserManagement adapter providing the following parameter mappings:

```
<param name="directoryType" value="ActiveDirectory"/>
<param name="mapAttribute" value="orclguid=objectguid"/>
<param name="mapAttribute" value="cn=sAMAccountName"/>
<param name="mapAttribute" value="uniquemember=member"/>
<param name="mapAttribute" value="OESRole=OESRole"/>
<param name="mapObjectclass" value="orclGroup=group"/>
<param name="mapObjectclass" value="groupofurls=group"/>
<param name="mapObjectclass" value="groupofuniquenames=group"/>
<param name="mapObjectclass" value="person=user"/>
<param name="mapRDNAttribute" value="uniquemember=member"/>
```

   For more information about configuring the UserManagement adapter, see the "UserManagement Plug-In" section in *Oracle Fusion Middleware Administrator's Guide for Oracle Virtual Directory*.

7. Add the `OES10gUserEntitlementsPlugin` and add all the plug-in parameters as shown in the example below, replacing the host and port details for BLM and Personalization (`p13n`) for your local environment:

```
<param name="ldap_group_basedn" value="cn=Groups,dc=us,dc=oracle,dc=com"/>
<param name="ldap_user_basedn" value="cn=Users,dc=us,dc=oracle,dc=com"/>
<param name="ldap_admin_user" value="cn=Administrator"/>
<param name="oes_admin_user" value="admin"/>
<param name="OrclOVDEncrypted_oes_admin_pass" value="<password>"/>
<param name="oes_config_name" value="JK_SM1"/>
<param name="oes_policy_domain" value="JK_SM1"/>
<param name="oes_resource_action" value="browse"/>
<param name="oes_resource_prefix" value="//app/policy/"/>
<param name="oes_resource_name"
value="javaapi_app/WebCenterApp/WebCenterResource"/>
<param name="oes_resource_namespace" value="webcenterResource"/>
<param name="oes_roles_cache_interval" value="180000"/>
<param name="oes_action_namespace" value="webcenterAction"/>
<param name="p13n_admin_user" value="weblogic"/>
<param name="oes_p13n_debug" value="true"/>
```

```
<param name="OrclOVDEncrypted_p13n_admin_pass" value="<password>"/>
<param name="oes_blm_host" value="example.com"/>
<param name="oes_blm_port" value="7011"/>
<param name="oes_p13n_index_url" value="example.com/>
<param name="oes_p13n_prop_url" value="example.com"/>
<param name="ldap_eqmatch" value="equalityMatch"/>
<param name="ldap_loginattr" value="sAMAccountName"/>
<param name="ldap_loginattr" value="mail"/>
<param name="ldap_loginattr" value="cn"/>
```

Note that passwords, once entered as plug-in parameters, are encrypted and then stored on the server.

8. Restart the OVD process using the following command:

```
./opmnctl stopall startall
```

Make sure that the OVD process restarts without any exceptions before continuing. If you encounter errors, you can turn on logging in the plug-in, by adding the following entry to:

ORACLE_INSTANCE_HOME`/config/OVD/ovd1/ovd-logging.xmlovd-logging.xml`

```
<logger name='oracle.webcenter.security.internal.plugins.ovd' level='TRACE:1'
useParentHandlers='false'>
      <handler name='OVDHandler'/>
</logger>
```

9. If SSL-enabled Personalization attributes are required, then import the certificate containing the public key of the Personalization server into the trust store on OVD, which is typically the JDK's `cacerts` file.

31.7.4.3 Configuring the Personalization Attributes

If you are using Personalization attributes as part of your constraints, then see the "Creating Property Sets and Property Definitions" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper* for how to configure them. For more information about Personalization, see Section 25, "Managing Personalization."

31.7.4.4 Configuring WebCenter Portal to Consume Dynamic Roles

This section describes how to prepare WebCenter Portal to consume dynamic roles defined in OES 10*g*.

By default, WebCenter Portal picks up only the static enterprise roles defined in the identity store. To use the dynamic roles defined within OES (Oracle Entitlements Server), you need to add the OVD plug-in as the authenticator. The OVD plug-in can then consolidate the static roles from the identity store and the dynamic roles from OES.

To configure WebCenter Portal to consume dynamic roles:

1. Log in to the WebLogic Server Administration Console.

   For information on logging into the WebLogic Server Administration Console, see Section 1.13.2, "Oracle WebLogic Server Administration Console."

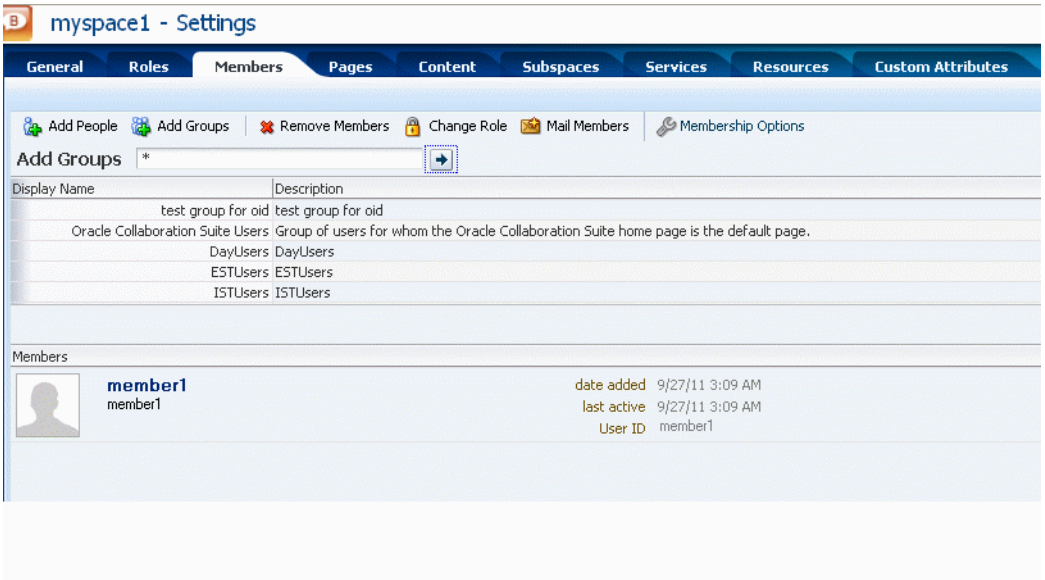
2. Add an authenticator of **Type** `Oracle Virtual Directory` providing the OVD connection details, and the group base `dn` and user base `dn`.

Leave the rest of the settings as their default values. Any directory-specific mapping should be done only in the adapter using the UserManagement plug-in. For more information about configuring the UserManagement adapter, see the "UserManagement Plug-In" section in *Oracle Fusion Middleware Administrator's Guide for Oracle Virtual Directory*.

3. Restart all servers.
4. Log onto WebCenter Portal as a user in OID and create a portal.
5. Go to **Add Members > Groups > Search** and add the enterprise roles you want as members to a portal as shown in Figure 31–23.

   With OVD as the authenticator, you should see both dynamic (from OES) and static groups. In Figure 31–23, the dynamic groups are `ESTUsers`, `DayUsers` and `ISTUsers`, with the rest being static groups from OID.

Figure 31–23 Adding Static and Dynamic Groups to a Portal



31.8 Configuring Dynamic Groups for WebCenter Portal

A dynamic group is a static group that is dynamically populated. Dynamic groups can be assigned to roles and used within WebCenter Portal in the same way as static groups.

Within the application, WebCenter Portal does not distinguish between static and dynamic groups. Dynamic groups are configured entirely in the identity store (and their configuration is specific to the LDAP implementation being used), and exposed in the same manner as static groups (in fact a dynamic group can be a composite of a static member list and a dynamically determined membership).

The dynamic membership of the group is defined by setting the group's `labeledURI` attribute with an appropriate LDAP query filter. The query filter defines the set of users that will define the membership of the group.

For Oracle Internet Directory, you can create a dynamic group with an LDIF file and using the `ldapadd` command, or using the Oracle Directory Services Manager (ODSM). These two options are described in the following subsections:

- Section 31.8.1, "Creating a Dynamic Group Using an LDIF File"

- Section 31.8.2, "Creating a Dynamic Group Using the Oracle Directory Services Manager"

31.8.1 Creating a Dynamic Group Using an LDIF File

To create the dynamic group using an LDIF file:

1. Create an LDIF file with a text editor. The following example shows how a dynamic group can be defined that represents all users under the default user search base, with the title of "Manager":

Example 31–1 Defining a Dynamic Group Using an LDIF File

```
dn:
cn=managers,cn=portal.070720.104824.056918000,cn=groups,dc=us,dc=oracle,dc=com
labelеduri: ldap://myserver.example.com:12061/cn=users,dc=us,dc=mybiz,dc=com
??sub?(title=Manager)
description: Dynamic Group of Managers
cn: Managers
orclisvisible: true
objectclass: orclDynamicGroup
objectclass: orclGroup
objectclass: top
objectclass: groupOfUniqueNames
displayname: Managers
owner: cn=fmwadmin,cn=users,dc=us,dc=mybiz,dc=com
```

Note: .The `labledURI` syntax for an LDAP URL is defined in RFC 2255 (`http://www.faqs.org/rfcs/rfc2255.html`). In the example above, it is representing a search for any entry under the DN `cn=users,dc=us,dc=mybiz,dc=com` with the attribute `title=Manager`. This is to be done on the server `myserver.example.com` at LDAP port `12061` and using a subtree ("sub") search.

A dynamic group can be defined on any attribute or condition that can be represented as an LDAP URL and defined in the `labeledURI` attribute. Dynamic groups can also be defined using the `ConnectBy` assertion, which is included in the `orclDynamicGroup` objectClass. Refer to *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory* for more information for this alternate approach.

2. Save the file, and then update the OID server by issuing the `ldapadd` command. For example:

Example 31–2 Updating OID Using the ldapadd Command

```
ldapadd -h myserver -p 12061 -D cn=fmwadmin -w mybiz1 -f managers.ldif -v
add labeledуri:
ldap://myserver.example.com:12061/cn=users,dc=us,dc=mybiz,dc=com??sub?(title=Ma
nager)
add description:
Dynamic Group of Managers
add cn:
Managers
add orclisvisible:
true
```

```
add objectclass:
orclDynamicGroup
orclGroup
top
groupOfUniqueNames
add displayname:
Managers
add owner:
cn=fmwadmin,cn=users,dc=us,dc=mybiz,dc=com
adding new entry
cn=managers,cn=portal.070720.104824.056918000,cn=groups,dc=us,dc=mybiz,dc=com
modify complete
```

31.8.2 Creating a Dynamic Group Using the Oracle Directory Services Manager

To create a dynamic group using ODSM:

1. Invoke Oracle Directory Services Manager (ODSM) and connect to the Oracle Internet Directory server.

   Refer to the "Using Oracle Directory Services Manager" chapter in *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory* for information on invoking and using the Oracle Directory Services Manager.

2. From the Go to list, select Data Browser.
3. Click the New Entry icon in the data browser.
4. Provide the DN and add the objectclasses `orclDynamicGroup` and `groupOfUniqueNames`.
5. On the Mandatory Properties tab, provide the CN attribute.
6. On the Optional Properties tab, provide the attributes for labeleduri.
7. Click OK to complete the definition of the dynamic group.

When you refresh the tree view you'll see the new group that you created. Note that group members will not be shown in ODSM.

31.9 Configuring the REST Service Identity Asserter

This section describes how to configure an identity asserter for the REST service. For the REST service, including REST service APIs, to be used with WebCenter Portal or Portal Framework applications requires that an identity asserter be configured for it in the WebCenter domain identity store. The following sections show how to configure OPSS Trust Service instances and identity asserters for Oracle WebLogic Server.

This section contains the following subsections:

- Section 31.9.1, "Understanding the REST Service Instance and Identity Asserter"
- Section 31.9.2, "Setting up the Client Application"
- Section 31.9.3, "Configuring the WLS Trust Service Asserter"

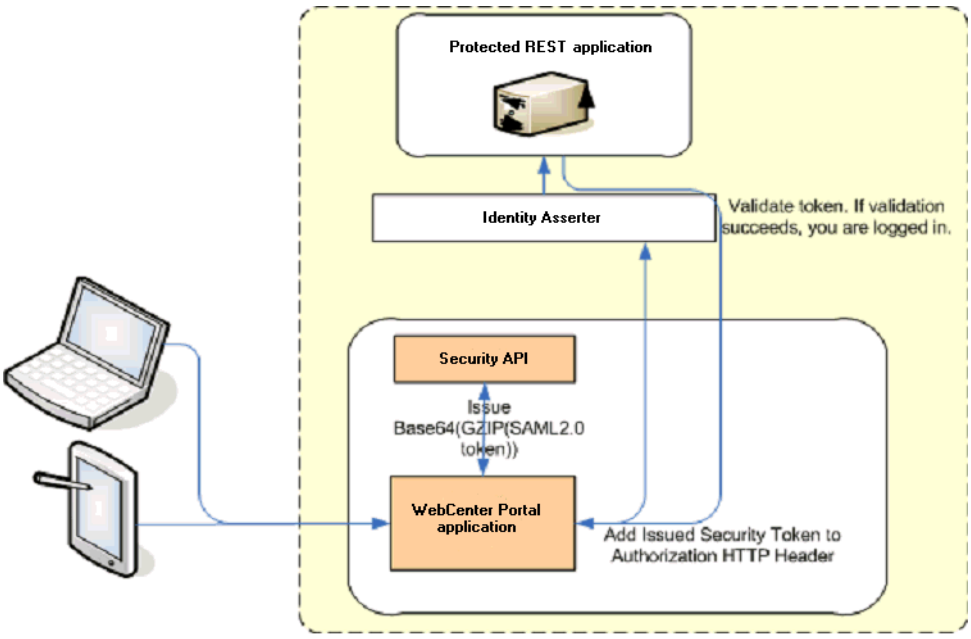
31.9.1 Understanding the REST Service Instance and Identity Asserter

Although WebCenter Portal and Portal Framework applications, and other Oracle WebLogic applications, can use REST APIs to display information the way they need to, since such calls originate from the mid-tier, users will be prompted again to provide login credentials. To overcome this, we use perimeter authentication where the user

identity is propagated in the HTTP header and asserted using the OPSS Trust Service Asserter.

In order to successfully propagate user identity from one application to another application, these applications must be using correctly configured Trust Service instances. Figure 31–24 shows the different components involved in the identity propagation and assertion.

Figure 31–24 REST Identity Propagation and Assertion



The following depicts the sequence of events involved in REST identity propagation and assertion:

1. End clients (browsers, smart phone apps) connect to a WebCenter Portal or Portal Framework application.
2. The application page queries data from REST APIs and builds its own UI on top and therefore needs to call the REST end point.
3. The application calls WebCenter Security API (`WCSecurityUtility.issueTrustServiceSecurityToken`) to issue the token used for securely propagating the user identity. The token is generated using the Trust Service Embedded Provider. Generated tokens are compressed to optimize token size and then BASE64-encoded to ensure that the token can be safely transported using an HTTP header.
4. The application takes the issued token and adds it against the "Authorization" security header. The client then dispatches the token as part of its call to the REST URI.
5. WebLogic Server checks if the identity asserter exists for the given token type.
6. The identity asserter parses and verifies that the token is using OPSS Trust Service APIs.
7. The asserter maps the username to a WLS username, a user Subject is established, and the call ends up on the REST application.

8. The REST application recognizes that the user is already an authenticated user and sends a response. The WebCenter Portal or Portal Framework application uses the response and shows the page to the end user.

31.9.2 Setting up the Client Application

This section describes how to configure the client for a REST service identity asserter.

To configure the client for a REST service identity asserter:

1. Using JDeveloper, create the client application.

   The client application could be a JSE or a servlet application. The following example shows the skeleton of a sample client application.

```
// The authenticated username
// String user = "weblogic";
// URL of the target application
URL url = "http://host:port/destinationApp";
//----------------------------------------

String b64EncodedToken = WCSecurityUtility.issueTrustServiceSecurityToken()

HttpURLConnection connection = (HttpURLConnection) url.openConnection();
connection.setRequestMethod("GET");
connection.setDoOutput(true);
connection.setReadTimeout(10000);
connection.setRequestProperty("Authorization", AUTH_TYPE_NAME + " " +
b64EncodedToken);
connection.connect();
BufferedReader rd = new BufferedReader(new InputStreamReader(
    connection.getInputStream()));
StringBuilder sb = new StringBuilder();

String line = null;
while ((line = rd.readLine()) != null) {
    sb.append(line);
}
connection.disconnect();
System.out.println(sb.toString());
```

2. Create and configure the keystore.

   Create the keystore for the domain and then configure WebLogic Server for the identity asserter. The keystore is first provisioned for a client certificate and private key. The client certificate is then exported and imported into a trust key store.

   a. Create the keystore as shown in Section 36.1.2.1, "Creating the WebCenter Portal Domain Keystore."

   b. Configure the keystore as shown in Section 36.1.2.2, "Configuring the Keystore with WLST," or Section 36.1.2.3, "Configuring the Keystore Using Fusion Middleware Control."

3. Edit the `jps-config.xml` configuration file.

   a. Navigate to your `DOMAIN_HOME/config/fmwconfig` directory and open the `jps-config.xml` file in a text editor.

   b. Make sure you have the following in the `jps-config.xml` file:

```
<serviceInstance name="keystore" provider="keystore.provider"
location="./default-keystore.jks">
```

c. Modify the `trust.provider.embedded` `propertySet` node as below:

```
<propertySets>
    <propertySet name="trust.provider.embedded">
         ... existing entries
        <property value="orakey" name="trust.aliasName"/>
        <property value="orakey" name="trust.issuerName"/>
    </propertySet>
</propertySets>
```

Where:

`trust.aliasName` is the alias looked up by the identity asserter in the configured keystore for a certificate with which the asserter verifies the issued trust token.

`trust.issuerName` is the alias looked up by the token issuer to look up the private key with which the trust token is issued/signed.

4. If the client and REST applications are in different domains, repeat these steps for both domains.
5. Restart all servers.

31.9.3 Configuring the WLS Trust Service Asserter

This section describes how to configure the WebLogic Server Trust Service asserter.

To configure the WebLogic Server Trust Service asserter:

1. Log into the WebLogic Administration Console as an administrator.
2. Navigate to **Security Realms -> myrealm**.
3. Open the Providers tab, and then the Authentication subtab.

   The Create a New Authentication Provider page displays.
4. Enter the **Name** of the new asserter (for example, `TrustServiceIdAsserter`).
5. Select `TrustServiceIdentityAsserter` as the asserter **Type**.

   This asserter calls the Trust Service APIs to decode and validate the token from the incoming request, and pass the username to the WebLogic for establishing the asserted subject.
6. Click **OK** to save your changes.
7. Restart all managed servers.

32

Configuring the Policy and Credential Store

For production environments, you must reassociate your policy store with an external LDAP (either Oracle Internet Directory 11gR1 or 10.1.4.3), or a database. Note that when using an external LDAP-based store, the credential store and policy store must be configured to use the same LDAP server. The identity store can, however, use any of the other supported LDAP servers; it does not need to use the same LDAP server as the policy and credential stores.

Reassociating the policy and credential store with OID consists of creating a root node in the LDAP directory, and then reassociating the policy and credential store with the OID server using Fusion Middleware Control, or from the command line using WLST. Reassociating the policy and credential store with a database consists of setting up the schema and database connection in the RCU, and then migrating the policy and credential store to the database from the command line using WLST. For troubleshooting information, see the "Reassociation Failure" section in the *Oracle Fusion Middleware Application Security Guide*.

Caution: Before reassociating the policy store, be sure to back up the relevant configuration files:

- `jps-config.xml`
- `system-jazn-data.xml`

As a precaution, you should also back up the `boot.properties` file for the Administration Server for the domain.

This chapter contains the following sections:

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console. Users with the `Monitor` or `Operator` roles can view security information but cannot make changes.

See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

32.1 Creating a root Node

The first step in reassociating the policy and credential store with OID, is to create an LDIF file in the LDAP directory and add a root node under which all data is added. To create the root node, follow the steps in the "Prerequisites to Using an LDAP-Based Security Store" section in the *Oracle Fusion Middleware Application Security Guide*. After creating the file and adding the node, continue by reassociating the store using either Fusion Middleware Control or WLST.

32.2 Reassociating the Credential and Policy Store Using Fusion Middleware Control

Before reassociating the policy and credential store with Oracle Internet Directory, you must first have created the root node as described in the "Prerequisites to Using an LDAP-Based Security Store" section in the *Oracle Fusion Middleware Application Security Guide*. After creating the root node, follow the steps in the "Reassociating with Fusion Middleware Control" section in the *Oracle Fusion Middleware Application Security Guide*. If the reassociation fails, see the "Reassociation Failure" section in the *Oracle Fusion Middleware Application Security Guide*.

32.3 Reassociating the Credential and Policy Store Using WLST

Before reassociating the policy and credential store with Oracle Internet Directory, you must first have created the root node as described in the "Prerequisites to Using an LDAP-Based Security Store" section in the *Oracle Fusion Middleware Application Security Guide*. If the reassociation fails, see the "Reassociation Failure" section in the *Oracle Fusion Middleware Application Security Guide*.

To reassociate the Credential and Policy Store using WLST:

1. Start WLST as described in Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."
2. Connect to the Administration Server for the target domain with the following command:

   ```
   connect('username>,'password', 'host_id:port')
   ```

   where:

   - *username* is the administrator account name used to access the Administration Server (for example, `weblogic`)
   - *password* is the administrator password used to access the Administration Server (for example, `weblogic`)
   - *host_id* is the server ID of the Administration Server (for example, `example.com`)

- *port* is the port number of the Administration Server (for example, 7001).

3. Reassociate the policy and credential store using the reassociateSecurityStore command:

```
reassociateSecurityStore(domain="domain_name", admin="admin_name",
password="password",
ldapurl="ldap_uri", servertype="ldap_srvr_type", jpsroot="root_webcenter_xxxx")
```

Where:

- *domain_name* specifies the domain name where reassociation takes place.
- *admin_name* specifies the administrator's user name on the LDAP server. The format is cn=usrName.
- *password* specifies the password associated with the user specified for the argument admin.
- *ldap_uri* specifies the URI of the LDAP server. The format is ldap://host:port, if you are using a default port, or ldaps://host:port, if you are using a secure LDAP port. The secure port must have been configured to handle an anonymous SSL connection, and it is distinct from the default (non-secure) port.
- *ldap_srvr_type* specifies the kind of the target LDAP server. Specify OID for Oracle Internet Directory.
- *root_webcenter_xxxx* specifies the root node in the target LDAP repository under which all data is migrated. Be sure to include the cn=. The format is cn=nodeName.

All arguments are required. For example:

```
reassociateSecurityStore(domain="myDomain", admin="cn=adminName",
password="myPass", ldapurl="ldaps://myhost.example.com:3060", servertype="OID",
jpsroot="cn=testNode")
```

32.4 Reassociating the Policy and Credential Store with a Database

As well as using an LDAP server, such as OID, for your policy and credential store, you can also reassociate the policy and credential store with an Oracle database. For the steps to configure a database as the Policy and Credential store, see the "Using a DB-Based OPSS Security Store" section in the *Oracle Fusion Middleware Application Security Guide*. If the reassociation fails, see the "Reassociation Failure" section in the *Oracle Fusion Middleware Application Security Guide*.

32.5 Managing Credentials

Administrators can manage credentials for the WebCenter Portal domain credential store using Fusion Middleware Control. For more information, see the "Managing Credentials with Fusion Middleware Control" section in the *Oracle Fusion Middleware Application Security Guide*.

32.6 Managing Users and Application Roles

This section describes how you can use Fusion Middleware Control, WLST, and the runtime administration pages in WebCenter Portal and Portal Framework applications to manage users and application roles.

This section contains the following subsections:

- Section 32.6.1, "Granting the WebCenter Portal Administrator Role"
- Section 32.6.2, "Granting Application Roles"
- Section 32.6.3, "Using the Runtime Administration Pages"

32.6.1 Granting the WebCenter Portal Administrator Role

WebCenter Portal only recognizes users in the identity store that is mapped by the first authenticator. Since the WebCenter Portal Administrator account is initially created only in the embedded LDAP server, if an external LDAP such as Oracle Internet Directory is configured as the primary authenticator for WebCenter Portal, you must also create a user in that LDAP and grant that user the WebCenter Portal Administrator role.

You can grant a user the WebCenter Portal Administrator role using Fusion Middleware Control or WLST as shown below in the sections on:

- Section 32.6.1.1, "Granting the WebCenter Portal Administrator Role Using Fusion Middleware Control"
- Section 32.6.1.2, "Granting the WebCenter Portal Administrator Role Using WLST"

For more information, see the "Granting Administrator Privileges to a Non-Default User" section in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*.

32.6.1.1 Granting the WebCenter Portal Administrator Role Using Fusion Middleware Control

This section describes how to grant the WebCenter Portal administrator role to a user account other than the default "weblogic" account.

To grant the WebCenter Portal Administrator role using Fusion Middleware Control:

1. Log into Fusion Middleware Control and navigate to the WebCenter Portal home page.

   See Section 6.2, "Navigating to the Home Page for WebCenter Portal."

2. From the WebCenter Portal menu, select **Security -> Application Roles**.

   The Application Roles page displays (see Figure 32–1).

Figure 32–1 Application Roles Page

3. Search for the WebCenter Portal Administrator role:

   a. Select **Select Application Stripe to Search**.

   b. Select `webcenter`.

   c. In the **Role Name** field, enter the following internal identifier for the Administrator role, and then click the **Search** (arrow) icon:

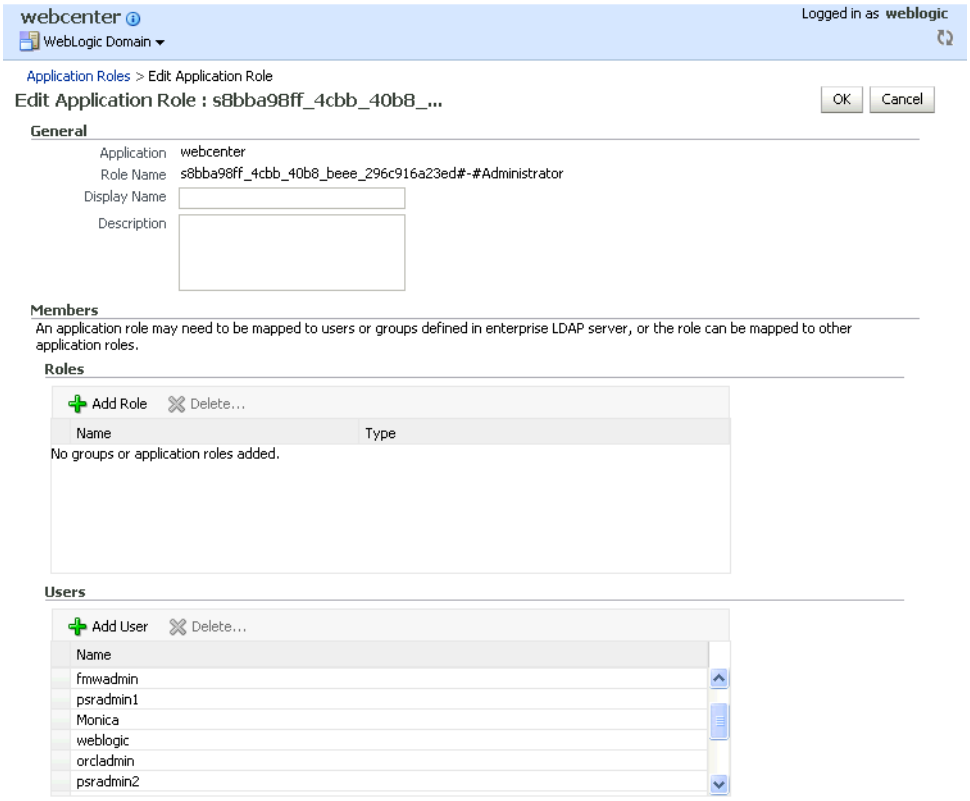
      ```
      s8bba98ff_4cbb_40b8_beee_296c916a23ed#-#Administrator
      ```

   The search should return `s8bba98ff_4cbb_40b8_beee_296c916a23ed#-#Administrator`, which is the administrator role identifier.

4. Click the administrator role identifier in the Role Name column.

   The Edit Application Role page displays (see Figure 32–2).

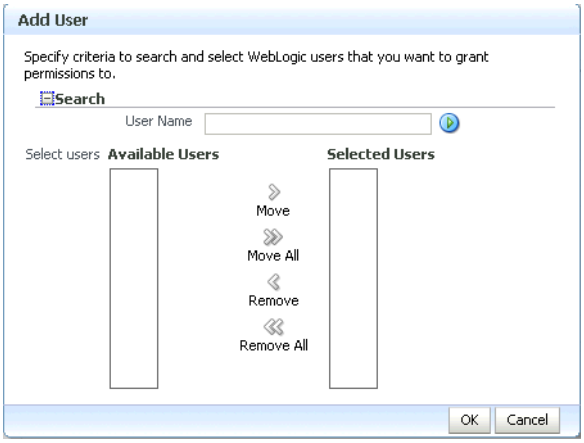
Figure 32–2 Edit Application Role Page



5. Click **Add User**.

   The Add User pop-up displays (see Figure 32–3).

Figure 32–3 Add User Pop-up



6. Use the Search function to search for the user to assign the Administrator role to.
7. Use the arrow keys to move the user from the Available Users column to the Selected Users column, and click **OK**.
8. On the Edit Application Role page, click **OK**.
9. To remove the weblogic role, on the Edit Application Role page under **Users**, click `weblogic` and the click **Delete**.

10. Restart the `WC_Spaces` managed server.

    When you login to WebCenter Portal, the Administration link should appear and you should be able to perform all administrator operations.

32.6.1.2 Granting the WebCenter Portal Administrator Role Using WLST

To grant the WebCenter Portal Administrator role to another user using WLST:

1. Start WLST as described in Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."
2. Connect to the WebCenter Portal Administration Server for the target domain with the following command:

   ```
   connect('user_name','password, 'host_id:port')
   ```

   Where:

   - *user_name* is the name of the user account with which to access the Administration Server (for example, `weblogic`)
   - *password* is the password with which to access the Administration Server
   - *host_id* is the host ID of the Administration Server
   - *port* is the port number of the Administration Server (for example, `7001`).
3. Grant the WebCenter Portal administrator application role to the user in Oracle Internet Directory using the `grantAppRole` command as shown below:

   ```
   grantAppRole(appStripe="webcenter",
   appRoleName="s8bba98ff_4cbb_40b8_beee_296c916a23ed#-#Administrator",
   principalClass="weblogic.security.principal.WLSUserImpl",
   principalName="wc_admin")
   ```

   Where *wc_admin* is the name of the administrator account to create.
4. To test the new account, log into WebCenter Portal using the new account name.

   The Administration link should appear, and you should be able to perform all administrator operations.
5. After granting the WebCenter Portal Administrator role to new accounts, remove this role from accounts that no longer need or require it using the WLST `revokeAppRole` command. For example, if WebCenter Portal was installed with a different administrator user name than `weblogic`, the administrator role should be given to that user and should be revoked from the default `weblogic`.

   ```
   revokeAppRole(appStripe="webcenter",
   appRoleName="s8bba98ff_4cbb_40b8_beee_296c916a23ed#-#Administrator",
   principalClass="weblogic.security.principal.WLSUserImpl",
   principalName="weblogic")
   ```

32.6.2 Granting Application Roles

This section describes how to add users to application roles using Fusion Middleware Control and WLST commands.

This section contains the following subsections:

- Section 32.6.2.1, "Granting Application Roles Using Fusion Middleware Control"
- Section 32.6.2.2, "Granting Application Roles Using WLST"

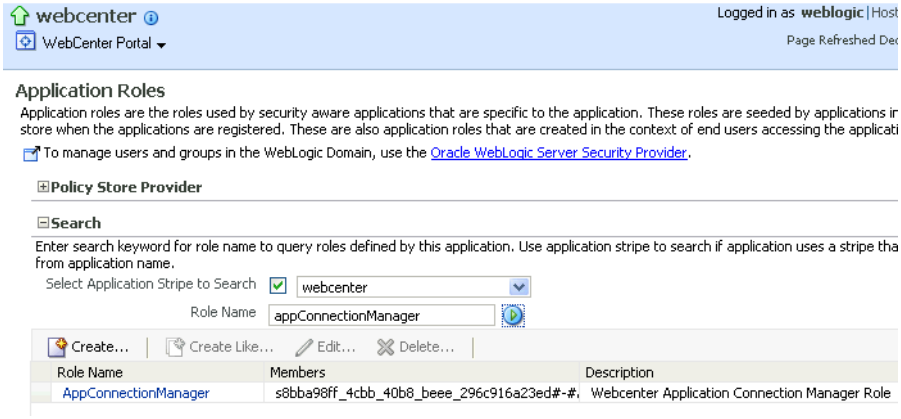
32.6.2.1 Granting Application Roles Using Fusion Middleware Control

This section describes how to grant an application role to users using Fusion Middleware Control.

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal or your Portal Framework application. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. From the WebCenter Portal menu, select **Security -> Application Roles**.

   The Application Roles page displays (see Figure 32–4).

Figure 32–4 Application Roles Page

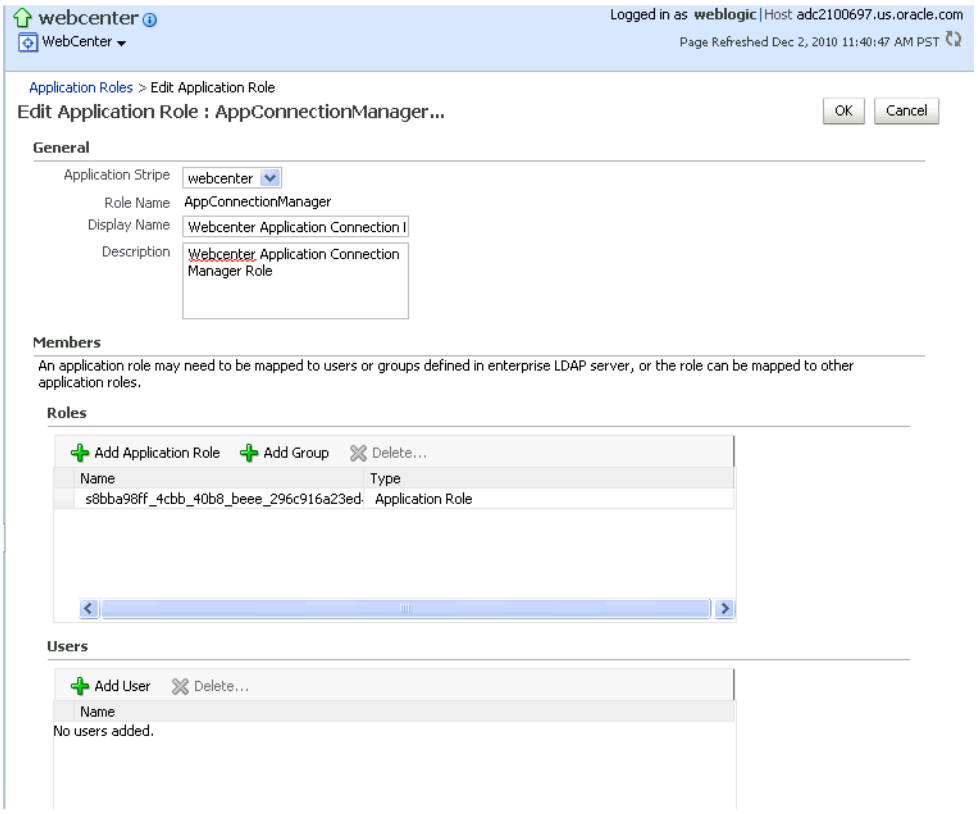


3. Search for the WebCenter Portal or Portal Framework application role:
   a. Select **Select Application Stripe to Search**.
   b. Sect the application stripe (`webcenter` for WebCenter Portal).
   c. In the **Role Name** field, enter the name of the role you are looking for (for example, `appConnectionManager`), and then click the **Search** (arrow) icon:

      If you are not sure of the name, enter a partial search term or leave the field blank to display all the application roles.
4. Click the role identifier in the Role Name column.

   The Edit Application Role page displays (see Figure 32–5).

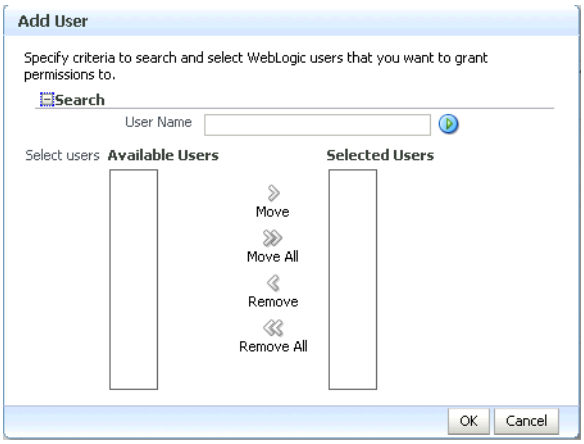
Figure 32–5 Edit Application Role Page



5. Click **Add User**.

   The Add User pop-up displays (see Figure 32–6).

Figure 32–6 Add User Pop-up



6. Use the Search function to search for the user to assign the application role to.
7. Use the arrow keys to move the user from the Available Users column to the Selected Users column, and click **OK**.
8. On the Edit Application Role page, click **OK**.
9. Restart the managed server on which WebCenter Portal or the Portal Framework application is deployed (for WebCenter Portal this is always `WC_Spaces`).

32.6.2.2 Granting Application Roles Using WLST

Use the `grantAppRole` command to grant an application role to a user. For syntax and usage information, see "grantAppRole" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

32.6.3 Using the Runtime Administration Pages

WebCenter Portal provides a *Security tab* from which an administrator can define application roles and grant application roles to users defined in the identity store. For information about managing users and application roles in WebCenter Portal, see the "Managing Users and Roles for WebCenter Portal" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

Caution: The "Allow Password Change" property, which specifies whether users can change their passwords within WebCenter Portal, should be carefully controlled for corporate identity stores. WebCenter Portal administrators can set this property from the Profile Management Settings page in WebCenter Portal. For more information, see the "Configuring Profiles" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

Portal Framework applications can provide a similar Security tab for application administrators. For details, see Section 43.4, "Managing Members and Roles for Portal Framework Applications." For more information about role-mapping for ADF-security based Portal Framework applications, see the "What You May Need to Know About Enterprise Roles and Application Roles" section in *Oracle Fusion Middleware Fusion Developer's Guide for Oracle Application Development Framework*.

32.7 Configuring Self-Registration By Invitation in WebCenter Portal

WebCenter Portal supports self-registration by invitation, as described in the "Enabling Self-Registration By Invitation-Only" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*. The self-registration 'by-invitation' feature requires that the WebCenter Portal domain credential store contain the following password credentials:

- `map name = o.webcenter.security.selfreg`
- `key= o.webcenter.security.selfreg.hmackey`
- `user name = o.webcenter.security.selfreg.hmackey`

To enable 'self-registration by invitation' in WebCenter Portal, use Fusion Middleware Control or the WLST command `createCred` to create the password credentials detailed above. For example:

```
createCred(map="o.webcenter.security.selfreg",
key="o.webcenter.security.selfreg.hmackey", type="PC",
user="o.webcenter.security.selfreg.hmackey", password="<password>", url="<url>",
port="<port>", [desc="<description>"])
```

For more information, see the "Managing Credentials with OPSS Scripts" section in the *Oracle Fusion Middleware Application Security Guide*.

32.8 Setting the Policy Store Refresh Interval and Other Cache Settings

This section provides recommended cache settings that should be configured after installation. Although settings for cache sizes and maximum group hierarchies should be based on your specific environment, the following sections provide recommendations that you can use as a starting point. For a complete list of tuning parameters and recommended values for WebCenter Portal, see the "Oracle WebCenter Portal Performance Tuning" section in the *Oracle Fusion Middleware Performance and Tuning Guide*.

This section includes the following subsections:

- Section 32.8.1, "Setting the Policy Store Refresh Interval"
- Section 32.8.2, "Setting the Connection Pool Cache"
- Section 32.8.3, "Setting User Cache Settings"
- Section 32.8.4, "Setting Group Cache Settings"

32.8.1 Setting the Policy Store Refresh Interval

The authorization policies used by WebCenter Portal use an in-memory cache with a default policy refresh time of 10 minutes. When a portal is created in a multi-node high availability environment, and you need a node failure to replicate the policy data more quickly, you can shorten the policy store refresh interval by modifying the domain-level `jps-config.xml` file, and adding the following entry:

```
oracle.security.jps.ldap.policystore.refresh.interval=<time_in_milli_seconds>
```

This should be added to the PDP service node:

```
<serviceInstance provider="pdp.service.provider" name="pdp.service">
```

Note that the policy refresh interval should not be set to too small a value as the frequency at which the server cached policy is refreshed may impact performance.

After modifying the `jps-config.xml` file, restart all servers in the domain. For more information, see the "Caching and Refreshing the Cache" section in the *Oracle Fusion Middleware Application Security Guide*.

32.8.2 Setting the Connection Pool Cache

This section describes the recommended settings for the connection pool cache.

To set the connection pool cache:

1. Log into the WLS Administration Console.
2. Select **Security Realms** > *[realm]* > **Providers** > *[provider]* > **Configuration** > **Provider Specific**.
3. Set the connection pool cache parameters to the following recommended values:
   - **Connection Pool Size** = max connection users
   - **Connect Timeout** = `30`
   - **Connection Retry Limit** = `1`
   - **Results Time Limit** = `1000`
   - **Keep Alive Enable** = `true`
4. Save your changes and restart all servers in the domain.

32.8.3 Setting User Cache Settings

This section describes the recommended settings for user cache settings.

To set user cache settings:

1. Log into the WLS Administration Console.
2. Select **Security Realms >** ***[realm]*** **> Providers >** ***[provider]*** **> Configuration > Provider Specific**.
3. Set the user cache parameters to the following recommended values:
   - **Cache Enabled** = `true`
   - **Cache Size** = `3200`
   - **Cache TTL** = `session timeout`
   - **Results Time Limit** = `1000`
   - **Keep Alive Enable** = `true`
4. Save your changes and restart all servers in the domain.

32.8.4 Setting Group Cache Settings

This section describes the recommended settings for group cache settings.

To set group cache settings:

1. Log into the WLS Administration Console.
2. Select **Security Realms >** ***[realm]*** **> Providers >** ***[provider]*** **> Performance**.
3. Set the group cache parameters to the following recommended values:
   - **Enable Group Membership Lookup Hierarchy Caching** = `true`
   - **Cache Size** = `3200`
   - **Max Group Hierarchies in Cache** = `1024`
   - **Group Hierarchy Cache TTL** = `session timeout`
   - **Keep Alive Enable** = `true`
4. Save your changes and restart all servers in the domain.

33 Configuring Single Sign-on

This chapter describes the available single sign-on (SSO) solutions for your WebCenter Portal or Portal Framework application to use, and how each is configured.

This chapter includes the following sections:

> **Permissions:** To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console. Users with the `Monitor` or `Operator` roles can view security information but cannot make changes.
>
> See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

33.1 Introduction to Single Sign-on

Single sign-on can be implemented for WebCenter Portal and Portal Framework applications using several solutions. This section describes their benefits and recommended application.

Oracle Access Manager (OAM), part of Oracle's enterprise class suite of products for identity management and security, provides a wide range of identity administration and security functions, including several single sign-on options for WebCenter Portal and Portal Framework applications. OAM (in particular, OAM 11g) is the recommended single sign-on solution for Oracle WebCenter Portal 11g installations.

For deployment environments that are already invested in Oracle 10g infrastructure, and where the Oracle Application Server Single Sign-On (OSSO) server is used as the primary SSO solution, WebCenter Portal 11g and Portal Framework applications can also be configured to use OSSO for single sign-on.

For non-production, development environments where you do not have an enterprise-class single sign-on infrastructure like Oracle Access Manager or Oracle SSO, and you only need to provide a single sign-on capability within WebCenter

Portal or Portal Framework application and associated Web tools like discussions, and worklist, you can configure a SAML-based SSO solution. If you need to provide single sign-on for other enterprise applications as well, this solution is not recommended.

If your enterprise uses Microsoft desktop logins that authenticate with a Microsoft domain controller with user accounts in Active Directory, then configuring SSO with Microsoft Clients may also be an option to consider.

33.2 Configuring Oracle Access Manager (OAM)

Oracle Access Manager (OAM) provides flexible and extensible authentication and authorization, and provides audit services. This section describes how to configure WebCenter Portal and Portal Framework applications for OAM single sign-on authentication, including how to configure the WebLogic server side and the WebCenter Portal or Portal Framework application as the partner application participating in SSO. Note that for Portal Framework applications some additional configurations are required, as described in Section 34.4, "Configuring Portal Framework and Portlet Producer Applications for OAM."

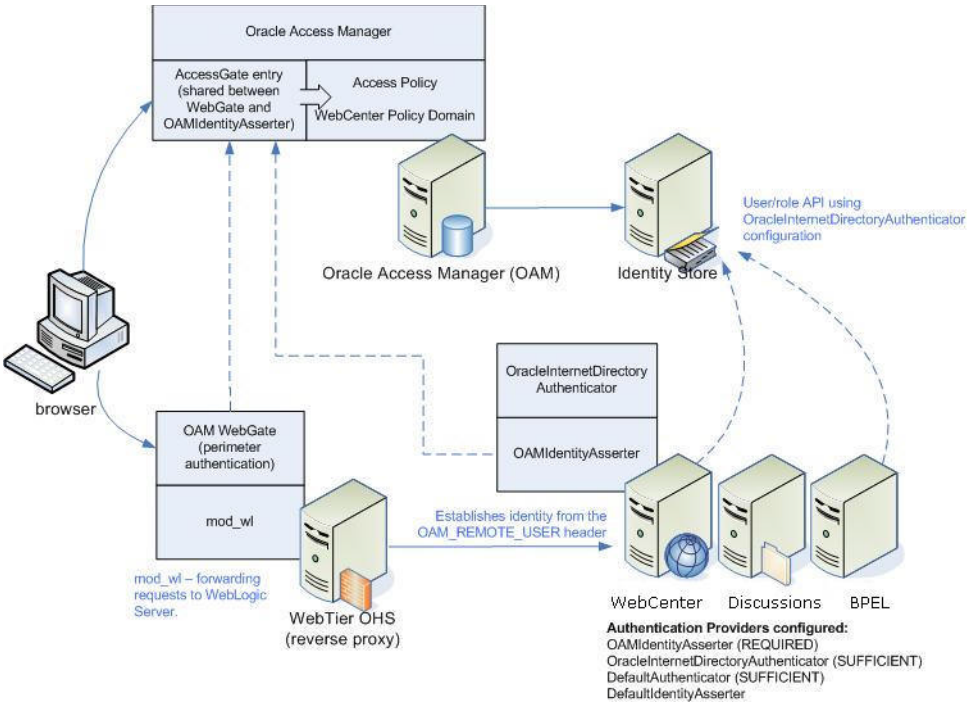
The installation and configuration steps for OAM 11*g* and 10*g* are presented in the following subsections:

- Section 33.2.1, "OAM Components and Topology"
- Section 33.2.2, "Roadmap to Configuring OAM"
- Section 33.2.3, "Installing and Configuring OAM"
- Section 33.2.4, "Configuring the WebLogic Domain for OAM"
- Section 33.2.5, "Installing and Configuring the Oracle HTTP Server"
- Section 33.2.6, "Additional Single Sign-on Configurations"
- Section 33.2.7, "Testing Your OAM Installation"

33.2.1 OAM Components and Topology

Figure 33–1 shows the components and topology required to set up single sign-on with Oracle Access Manager for a WebCenter Portal or Portal Framework application.

Figure 33–1 OAM Single Sign-On Components and Topology



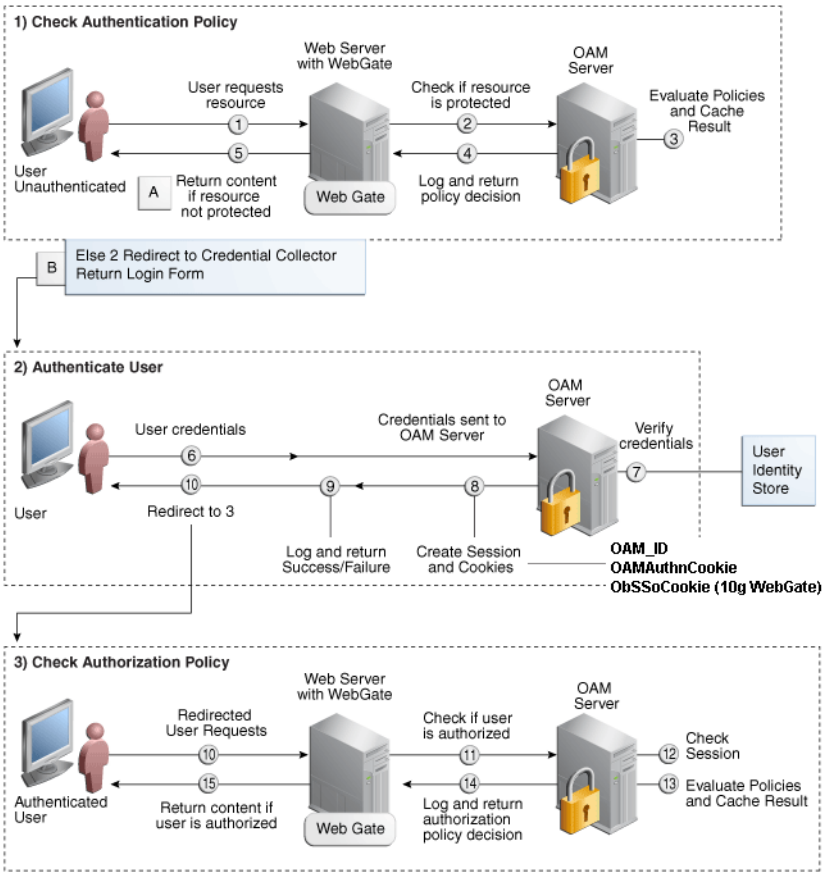
OAM consists of the following components:

- **Access Server** - a standalone server that provides authentication, authorization, and auditing services for Access Gates. There is one access server set up on OAM. This is done as part of the OAM install itself.
- **WebGate** - an out-of-the-box plugin that intercepts Web resource (HTTP) requests and forwards them to the Access Server for authentication and authorization.
- **Identity Assertion Provider (IAP)** - a type of security provider that asserts the identity of the user based on header information that is set by perimeter authentication. The OAM integration provides an OAM ID Asserter that can be configured as the OAM IAP. The OAM ID Asserter can be used for authentication or for identity assertion. For OAM SSO integration, the OAM ID Asserter should be configured as an Identity Assertion Provider (IAP) by selecting `obSSOCookie` under **Active Types** in the provider's Common settings.

OAM Single Sign-on Process Flow

Figure 33–2 shows the single sign-on process flow for OAM.

Figure 33–2 OAM Single Sign-on Process Flow



SSO Log-in Processing with OAM Agents

1. The user requests a resource.
2. The WebGate forwards the request to OAM for policy evaluation.
3. OAM:
   - Checks for the existence of an SSO cookie.
   - Checks policies to determine if the resource protected and if so, how?
4. The OAM server logs and returns decisions.
5. WebGate responds as follows:
   - Unprotected resource: resource is served to the user.
   - Protected resource:
     - Request is redirected to the credential collector
     - The login form is served based on the authentication policy
     - Authentication processing begins
6. User sends credentials.
7. OAM verifies credentials.
8. OAM starts the session and creates the following host-based cookies:

- One per partner: `OAMAuthnCookie` set by 11g WebGates (`ObSSOCookie` set by 10g WebGate) using the authentication token received from the OAM server after successful authentication.

  Note: A valid cookie is required for a session.

- One for OAM Server: OAM_ID

9. OAM logs Success or Failure.
10. OAM Credential collector redirects to WebGate and authorization processing begins.
11. WebGate prompts OAM to look up policies, compare them to the user's identity, and determine the user's level of authorization.
12. OAM logs policy decision and checks the session cookie.
13. OAM Server evaluates authorization policies and cache the result.
14. OAM Server logs and returns decisions
15. WebGate responds as follows:
    - If the authorization policy allows access, the request get redirected to `mod_wl` which in turn redirects the request to the WLS server where the WebCenter Portal or Portal Framework application is running, and from where desired content or applications are served to the user, as shown below:

      WebGate -> mod_wl -> WebCenter Portal or Portal Framework application [, discussions, .. etc] --> Content is served to the authenticated user

    - If the authorization policy denies access, the user is redirected to another URL determined by the administrator.

33.2.2 Roadmap to Configuring OAM

The flow chart (Figure 33–3) and table (Table 33–1) in this section provide an overview of the prerequisites and tasks required to configure single sign-on for WebCenter Portal or Portal Framework application using OAM.

Figure 33–3 Configuring Single Sign-on for WebCenter Portal Using OAM

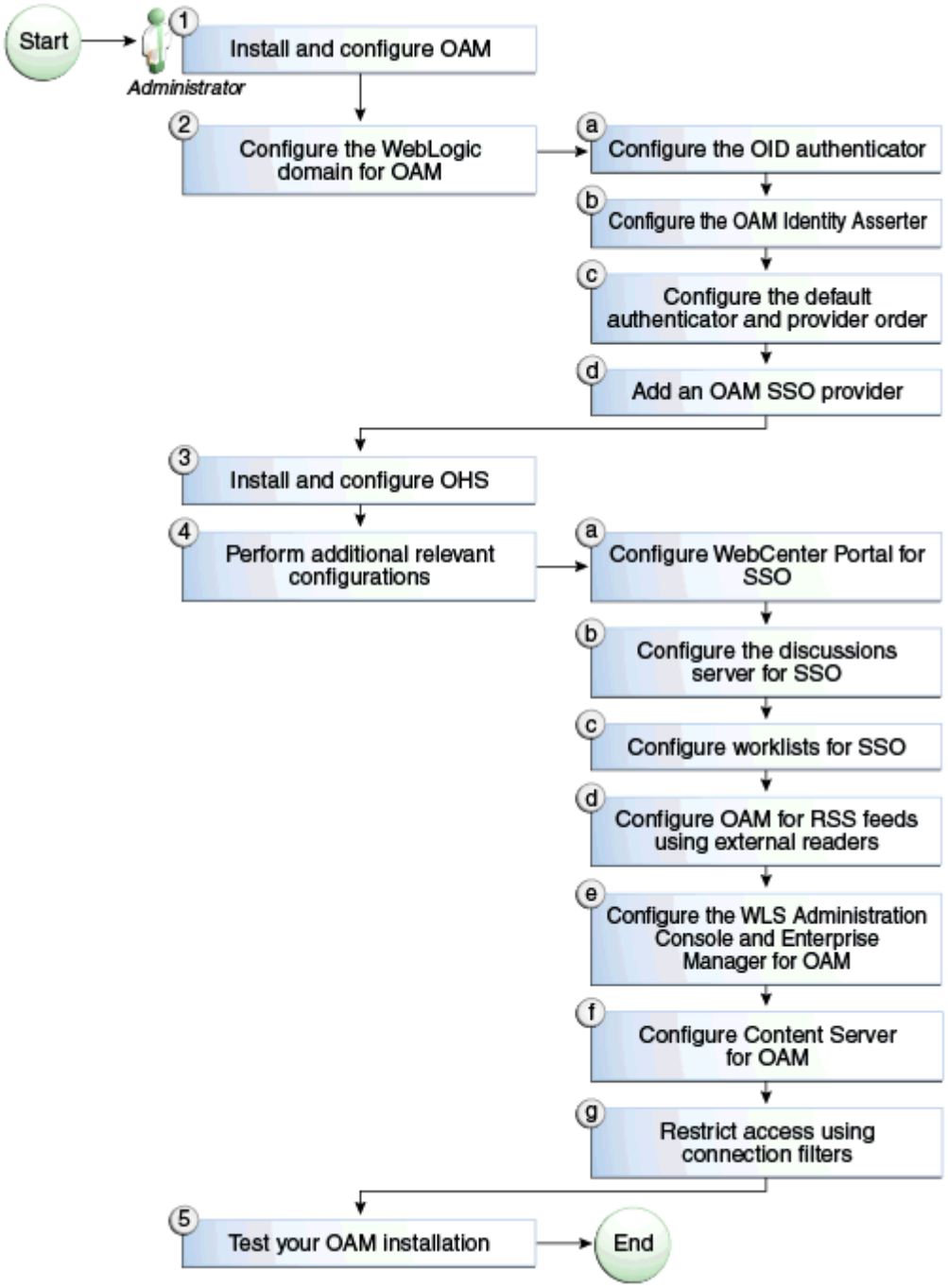


Table 33–1 shows the tasks and sub-tasks for configuring single sign-on for WebCenter Portal using OAM.

Table 33–1 Configuring Single Sign-on for WebCenter Portal Using OAM

Actor	Task	Sub-task	Notes
Administrator	**1.** Install and Configure OAM		Install and configure OAM 10g or 11g.
	2. Configure the WebLogic domain for OAM	**2.a** Configure the OID authenticator	
		2.b Configure the OAM identity asserter	
		2.c Configure the default authenticator and provider order	
		2.d Add an OAM SSO provider	
	3. Install and configure OHS		
	4. Perform additional configurations as required	**4.a** Configure WebCenter Portal for SSO	
		4.b Configure the discussions server for SSO	
		4.c Configure the worklists for SSO	
		4.d Configure OAM for RSS feeds using external readers	
		4.e Configure the WLS Administration Console and Enterprise Manager for OAM 11g or OAM 10g	
		4.f Configure Content Server for OAM	
		4.g Restrict access using connection filters	
	5. Test your OAM installation		

33.2.3 Installing and Configuring OAM

This section describes how to install and configure either OAM 11g or OAM 10g, the recommended single sign-on solutions for WebCenter Portal and Portal Framework applications.

> **Note:** Installing OAM should be performed only after you've installed Oracle WebCenter Portal (described in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*) and any other components required for your environment. You should also have configured and tested any required connections.

This section includes the following subsections:

- Section 33.2.3.1, "Installing and Configuring OAM 11g"
- Section 33.2.3.2, "Installing and Configuring OAM 10g"

33.2.3.1 Installing and Configuring OAM 11g

This section describes how to install and configure OAM 11g, and includes the following subsections:

- Section 33.2.3.1.1, "Installing and Configuring OAM 11g"
- Section 33.2.3.1.2, "Installing and Configuring the Oracle HTTP Server"
- Section 33.2.3.1.3, "Installing the WebGate on the WebTier"

- Section 33.2.3.1.4, "Registering the WebGate Agent"

33.2.3.1.1 Installing and Configuring OAM 11g

Install Oracle Access Manager (OAM) as described in the "Installing and Configuring Oracle Identity Management (11.1.1.7.0)" section in *Oracle Fusion Middleware Installation Guide for Oracle Identity Management*. Ideally, OAM and all the applications that participate in single sign-on should share the same identity store. By default, OAM uses the embedded LDAP identity store.

To configure OAM to use an external identity store, such as OID, see the "Registering a New User Identity Store" section in *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager with Oracle Security Token Service*. This section has pointers to setting the external identity store configured as the default or system store and configuring one or more authentication modules to point to this store. By default, the WebCenter policy configured in OAM uses the default authentication scheme (typically, the form-based authentication scheme `LDAPScheme`) specified in OAM,. If you intend to use the default scheme, the authentication module used by the scheme must point to the same identity store as your WebCenter installation. Optionally, you can choose to configure a different authentication scheme rather than the default, in which case you must also ensure that it points to the identity store used by WebCenter. Continue by configuring Oracle Access Manager in a WebLogic administration domain as described in the "Installing and Configuring Oracle Identity Management (11.1.1.7.0)" section in *Oracle Fusion Middleware Installation Guide for Oracle Identity Management*.

33.2.3.1.2 Installing and Configuring the Oracle HTTP Server

If you don't already have Oracle HTTP Server (OHS) installed, install OHS (11.1.1.4.0) as described in Section 33.2.5, "Installing and Configuring the Oracle HTTP Server."

If you do have an existing installation, you will need to apply a patch to bring it up to OHS (11.1.1.4.0) as described in the "Applying the Latest Oracle Fusion Middleware Patch Set" section in the *Oracle Fusion Middleware Patching Guide*.

After installing or patching OHS, continue by installing the WebGate as described in Section 33.2.3.1.3, "Installing the WebGate on the WebTier."

33.2.3.1.3 Installing the WebGate on the WebTier

This section describes how to install and configure the OHS WebGate.

> **Note:** Ensure that your Oracle HTTP server is down while installing OHS WebGate, and restart it only after you register the WebGate agent as described in Section 33.2.3.1.4, "Registering the WebGate Agent."

1. Install the WebGate as described in the "Installing and Configuring Oracle HTTP Server 11g WebGate for OAM" section in *Oracle Fusion Middleware Installation Guide for Oracle Identity Management*. Use the same middleware home that was specified during OHS install.
2. After installing Oracle HTTP Server 11g WebGate for Oracle Access Manager, move to the following directory under your Oracle Home for Webgate:

   For Unix operating systems:

   ```
   <Webgate_Home>/webgate/ohs/tools/deployWebGate
   ```

   For Windows operating systems:

```
<Webgate_Home>\webgate\ohs\tools\deployWebGate
```

3. From the command line, run the following command to copy the required bits of the agent from the `Webgate_Home` directory to the WebGate instance location:

   For Unix operating systems:

   ```
   ./deployWebGateInstance.sh -w <Webgate_Instance_Directory> -oh
   <Webgate_Oracle_Home>
   ```

   For Windows operating systems:

   ```
   deployWebGateInstance.bat -w <Webgate_Instance_Directory> -oh
   <Webgate_Oracle_Home>
   ```

   Where *<Webgate_Oracle_Home>* is the directory where you have installed Oracle HTTP Server WebGate and defined it as the Oracle Home for WebGate, as in the following example:

   ```
   <MW_HOME>/Oracle_OAMWebGate1
   ```

   The *<Webgate_Instance_Directory>* is the location of the Webgate Instance Home (which should be the same as the Instance Home of Oracle HTTP Server), as in the following example:

   ```
   <MW_HOME>/Oracle_WT1/instances/instance1/config/OHS/ohs1
   ```

   Note that an Instance Home for Oracle HTTP Server is created after you configure the Oracle HTTP Server. This configuration should be performed after installing or patching to Oracle HTTP Server 11.1.1.4.0.

4. Run the following command to ensure that the `LD_LIBRARY_PATH` variable contains *<Oracle_Home_for_Oracle_HTTP_Server>*`/lib`:

   For Unix operating systems (depending on the shell):

   ```
   export
   LD_LIBRARY_PATH=$LD_LIBRARY_PATH:<Oracle_Home_for_Oracle_HTTP_Server>/lib
   ```

   For Windows operating systems:

   Add the *<Webgate_Installation_Directory>*`\webgate\ohs\lib` and *<Oracle_Home_for_Oracle_HTTP_Server>*`\bin` locations to the `PATH` environment variable. Add a semicolon (;) followed by this path at the end of the entry for the PATH environment variable.

5. From your current working directory, move up one level:

   For Unix operating systems, move to:

   ```
   <Webgate_Home>/webgate/ohs/tools/setup/InstallTools
   ```

   For Windows operating systems, move to:

   ```
   <Webgate_Home>\webgate\ohs\tools\EditHttpConf
   ```

6. From the command line, run the following command to copy the `apache_webgate.template` from the `Webgate_Home` directory to the WebGate Instance location (renaming it to `webgate.conf`) and update the `httpd.conf` file to add one line to include the name of `webgate.conf` file:

   For Unix operating systems:

   ```
   ./EditHttpConf -w <Webgate_Instance_Directory> [-oh <Webgate_Oracle_Home>] [-o
   ```

```
<output_file>]
```

For Windows operating systems:

```
EditHttpConf.exe -w <Webgate_Instance_Directory> [-oh <Webgate_Oracle_Home>]
[-o <output_file>]
```

> **Note:** The `-oh <WebGate_Oracle_Home>` and `-o <output_file>` parameters are optional.

Where `<Webgate_Oracle_Home>` is the directory where you have installed Oracle HTTP Server WebGate and defined it as the Oracle Home for WebGate, as in the following example:

```
<MW_HOME>/Oracle_OAMWebGate1
```

The `<Webgate_Instance_Directory>` is the location of the Web Gate instance home (which should be the same as the instance home of OHS), as in the following example:

```
<MW_HOME>/Oracle_WT1/instances/instance1/config/OHS/ohs1
```

33.2.3.1.4 Registering the WebGate Agent

After installing the WebGate on the WebTier, you also need to register the WebGate agent. The steps below will automatically create a protected policy that uses the default Authentication Scheme that is configured in your OAM installation (typically, the form-based authentication scheme `LDAPScheme`). If you want to customize the single sign-on login page, or want resources to be protected by some other authentication scheme, then change it using the OAM Console (see the "Managing Authentication Schemes" chapter in *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager with Oracle Security Token Service* for more information).

> **Note:** If you are using WebCenter Portal in conjunction with other applications in your environment, and you require single sign-on for these applications, you must ensure that the authentication schemes used by these applications are either the same or at least at the same level and point to the same identity store.

For more information about registering the WebGate agent, see also "Getting Started with a New Oracle HTTP Server 11g WebGate Agent for Oracle Access Manager" in the Oracle Fusion Middleware Installing Webgates for Oracle Access Manager.

Follow the steps below to register the WebGate agent on the machine where OAM is installed using the `oamreg` tool in inband mode:

1. Change directories to `<RREG_Home>/input` (where `<RREG_Home>` is the directory to where you extracted the contents of `RREG.tar.gz/rreg`).
2. Copy over `$WEBCENTER_HOME/webcenter/scripts/webcenter.oam.conf` from the Oracle WebCenter Portal installation here.

   The default location for `WEBCENTER_HOME` is `$ORACLE_HOME/Oracle_WC1`.
3. Copy over `$SOA_HOME/soa/prov/soa.oam.conf` and `$WC_CONTENT_ORACLE_HOME/common/security/oam.conf` from the SOA and Content Server installations respectively.

The default location for SOA_HOME is $ORACLE_HOME/Oracle_SOA1 and the default location for WC_CONTENT_ORACLE_HOME is $ORACLE_HOME/Oracle_ECM1. Note that the SOA-related location mappings contained in soa.oam.conf only come into effect when deploying and using WebCenter Portal-provided workflows on a SOA server, and that even the SOA related URLS protected within webcenter.oam.conf will come into effect if SOA is being used.

4. Create a new file named WebCenterOAM11gRequest.xml to serve as a parameter file to the oamreg tool.

   In the example below, replace the contents within $$webtier..$$ with your WebTier host and port IDs, and $$oam...$$ with the OAM host and administration server port.

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

<!--
 Copyright (c) 2009, 2010, Oracle and/or its affiliates. All rights reserved.

   NAME: OAM11GRequest_short.xml - Template for OAM 11G Agent Registration
Request file
 (Shorter version - Only mandatory values - Default values will be used for all
other fields)
   DESCRIPTION: Modify with specific values and pass file as input to the tool.
-->
<OAM11GRegRequest>
    <serverAddress>http://$$oamhost$$:$$oamadminserverport$$</serverAddress>
    <hostIdentifier>$$webtierhost$$_webcenter</hostIdentifier>
    <agentName>$$webtierhost$$_webcenter</agentName>
    <logOutUrls>
        <url>/oamsso/logout.html</url>
    </logOutUrls>
</OAM11GRegRequest>
```

5. Change directories to *<RREG_Home>*.

6. Run the following command:

```
<RREG_Home>/bin/oamreg.sh inband input/WebCenterOAM11gRequest.xml
```

   - When prompted for agent credentials enter your OAM administrator credentials.
   - Enter your WebGate password.
   - Enter yes when asked whether you want to import a URIs file. Specify the full path to the <RREG_HOME>/input/webcenter.oam.conf file you copied there earlier.

     You should see output like that below indicating that registration has been successful:

```
----------------------------------------
Request summary:
OAM11G Agent Name:example_webcenter
URL String:example_webcenter
Registering in Mode:inband
Your registration request is being been sent to the Admin server at:
http://example.com:7001
----------------------------------------
Inband registration process completed successfully! Output artifacts are
```

```
created in the output folder.
```

7. Copy the generated files and artifacts (`ObAccessClient.xml` and `cwallet.sso`) from *<RREG_Home>*`/output/`*$$webtierhost$$*`_webcenter` to your WebGate instance configuration directory (`<Webgate_Instance_Directory>/webgate/config`). Note that `<Webgate_Instance_Directory>` should match the instance home of OHS, as in the following example:

   ```
   <MW_HOME>/Oracle_WT1/instances/instance1/config/OHS/ohs1/webgate/config
   ```

8. Change directories to *<RREG_Home>*`/input`.

9. If you have SOA or WebCenter Content Server installed

   a. Create a policy update file called `WebCenterOAM11gPolicyUpdate.xml` as shown in the example below, replacing the contents within `$$webtier..$$` with your WebTier host and port IDs, and `$$oam...$$` with the OAM host and administration server port as you did earlier:

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

   <!--
    Copyright (c) 2009, 2011, Oracle and/or its affiliates. All rights
   reserved.

      NAME: UpdatePolicyRequest.xml - Template for updating application domain
   and/or policies without changes to any agent profile
      DESCRIPTION: Modify with specific values and pass file as input to the
   tool
   -->
   <PolicyRegRequest>


   <serverAddress>http://$$oamhost$$:$$oamadminserverport$$</serverAddress>
       <hostIdentifier>$$webtierhost$$_webcenter</hostIdentifier>

   <applicationDomainName>$$webtierhost$$_webcenter</applicationDomainName>

   </PolicyRegRequest>
   ```

   b. Run the following command:

   ```
   <RREG_Home>/bin/oamreg.sh policyUpdate
   input/WebCenterOAM11gPolicyUpdate.xml
   ```

   Enter your OAM credentials when prompted. Enter `yes` when asked whether you want to import a URIs file, and specify `<RREG_HOME>/input/soa.oam.conf`.

   Your policy will be updated with SOA resources.

   c. Run the `policyUpdate` command again, this time specifying `<RREG_HOME>/input/oam.conf` to update the policy with Content Server resources. Your policy now contains Oracle WebCenter Portal, SOA and Content Server artifacts.

10. From the OAM Console, you should now be able to see the following artifacts:

    - 11g WebGate agent named `$$webtierhost$$_webcenter`
    - 11g host identifier by the same name

- an application domain with the same name containing authentication and authorization policies which in turn contain protected and public policies

11. Go to **Application Domain> $$webtierhost$$_webcenter > Authentication Policies**. You should be able to see the following policies:
    - Exclusion Scheme
    - Protected Resource Policy
    - Public Resource Policy
    - WebCenter REST Policy
12. Open the WebCenter REST Policy and make sure that the Authentication Scheme is set to `BasicSessionlessScheme` or `BasicScheme`.
13. Open the Resources tab and search for resources with their Authentication Policy set to `Exclusion Scheme`. You should see the following resources:
    - `/rsscrawl*`
    - `/rsscrawl/.../*`
    - `/sesUserAuth*`
    - `/sesUserAuth/.../*`
    - `/services-producer/portlets*`
    - `/services-producer/portlets/.../*`
    - `/wsrp-tools/portlets`
    - `/wsrp-tools/portlets/.../*`
14. Select the `/rsscrawl*` resource in the search results and click Edit.
15. Change the Protection Level from `Protected` to `Excluded` and click **Apply**. Note that the resource's authentication policy and authorization policy is removed.
16. Close the Resources tab and repeat the steps for the remaining `Exclusion Scheme` resources.

    When you now search for resources with their Authentication Policy set to `Exclusion Scheme` you should see no results.
17. Restart OHS.
18. After installing and configuring the WebTier and associated components, continue by configuring the Policy Manager as described in Section 33.2.4, "Configuring the WebLogic Domain for OAM," and performing any additional service and component configurations that apply as described in Section 33.2.6, "Additional Single Sign-on Configurations."

33.2.3.2 Installing and Configuring OAM 10g

This section describes how to install and configure OAM 10g, and includes the following subsections:

- Section 33.2.3.2.1, "Installing and Configuring OAM 10g"
- Section 33.2.3.2.2, "Installing and Configuring the Oracle HTTP Server"
- Section 33.2.3.2.3, "Configuring the WebCenter Portal Policy Domain"
- Section 33.2.3.2.4, "Installing the WebGate 10g on the WebTier"

33.2.3.2.1 Installing and Configuring OAM 10g

If you don't already have Oracle Access Manager (OAM) 10*g* installed, install OAM 10*g* as described in the *Oracle Access Manager Installation Guide*.

33.2.3.2.2 Installing and Configuring the Oracle HTTP Server

If you don't already have Oracle HTTP Server (OHS) installed, install OHS (11.1.1.4.0) as described in Section 33.2.5, "Installing and Configuring the Oracle HTTP Server."

If you do have an existing installation, you will need to apply a patch to bring it up to OHS (11.1.1.4.0) as described in the "Applying the Latest Oracle Fusion Middleware Patch Set" section in the *Oracle Fusion Middleware Patching Guide*.

After installing or patching OHS, continue by installing the WebGate as described in Section 33.2.3.2.3, "Configuring the WebCenter Portal Policy Domain."

33.2.3.2.3 Configuring the WebCenter Portal Policy Domain

These steps assume that you've installed Oracle WebCenter Portal. By default, an Oracle WebCenter Portal installation creates a WebLogic Server domain, including an Administration Server and four managed servers: `WC_Spaces`, `WC_Collaboration`, `WC_Utilities`, and `WC_Portlet`.

1. Determine which access server to use.
   a. Log onto the Access Manager.
   b. Click **Access System Console**.
   c. Open the Access System Configuration tab.
   d. Click **Access Server Configuration** to display a list of all access servers.
   e. Click an access server in the list to see server details.

      The host name and port are the values you need for the `oam_aaa_host` and `oam_aaa_port` parameters respectively in the script.
2. Check that `OraDefaultExclusionAuthNScheme` is available in your OAM 10g installation. If it does not exist, create the OraDefaultExclusionAuthNScheme as shown below:
   a. Open the OAM Access System Console.
   b. Click Authentication Management.
   c. Click Add.
   d. Specify `OraDefaultExclusionAuthNScheme` in the Name field.
   e. Enter `To exclude resources from being protected by OAM` in the Description field.
   f. Enter `0` in the Level field.
   g. Specify `None` in the Challenge Method field.
   h. Add `unprotected:true` to the Challenge Parameter field.
   i. Click Save.
   j. Open the Plugins tab for this authentication scheme and click Modify.
   k. Select `credential_mapping` from the drop down list.
   l. Specify a value as:

```
obMappingBase="dc=us,dc=oracle,dc=com",obMappingFilter="(u
id=OblixAnonymous)"
```

Make sure that this value matches the corresponding field for the `OraDefaultAnonAuthNScheme`.

m. Click Save.

n. Open the General tab again and click Modify.

o. Check `Yes` for Enabled.

p. Click Save.

3. Run the following command.

The `oamcfgtool.jar` is available in `ORACLE_HOME/modules/oracle.oamprovider_11.1.1/oamcfgtool.jar` in the WebCenter Portal installation. Values in bold are the ones that you must supply based on the settings of your WebCenter Portal and OAM instances.

```
java -jar ORACLE_HOME/modules/oracle.oamprovider_11.1.1/oamcfgtool.jar
mode=CREATE app_domain=<your_domain_name>
uris_file=WEBCENTER_HOME/webcenter/scripts/webcenter.oam.conf"
app_agent_password=<Password to be provisioned for App Agent>
ldap_host=<Hostname of LDAP server> ldap_port=<Port of LDAP server>
ldap_userdn=<DN of LDAP Admin User, usually "cn=orcladmin">
ldap_userpassword=<Password of LDAP Admin User>
oam_aaa_host=<HOST of OAM server> oam_aaa_port=<Port of OAM server>
```

We recommend that you register your domain (for <**your_domain_name**>) as something like "`webtier.example.com`", where "`webtier.example.com`" is your WebTier, so that you can easily distinguish the various policies in OAM.

If your command ran successfully, you should see something like the following output depending on the values you used:

```
Processed input parameters
Initialized Global Configuration
Successfully completed the Create operation.
Operation Summary:
Policy Domain : webtier.example.com
Host Identifier: webtier.example.com
Access Gate ID : webtier.example.com_AG
```

You can also run the Validate command to validate your configurations:

```
java -jar WCP_ORACLE_HOME/modules/oracle.oamprovider_11.1.1/oamcfgtool.jar
mode=VALIDATE app_domain=<your_domain_name>
ldap_host=<Hostname of LDAP server> ldap_port=<Port of LDAP server>
*ldap_userdn=<DN of LDAP Admin User, usually "cn=orcladmin">*
ldap_userpassword=<Password of LDAP Admin User>
oam_aaa_host=<HOST of OAM server>
oam_aaa_port=<Port of OAM server>
test_username=<Username to be used for policy validation>
test_userpassword=<Userpassword to be used for policy validation>
```

If your command runs successfully, you should see the same output as above.

4. If your instance also contains a SOA installation, then run `oamcfgtool` again to protect the SOA URIs in the policy domain you created in the previous step. Use the same parameters as the ones used in the previous step so that the existing policy domain is updated with URIs in the `soa.oam.conf file`.

```
java -jar ORACLE_HOME/modules/oracle.oamprovider_11.1.1/oamcfgtool.jar
mode=CREATE app_domain=<your_domain_name>
uris_file="SOA_HOME/soa/prov/soa.oam.conf"
app_agent_password=<Password to be provisioned for App Agent>
ldap_host=<Hostname of LDAP server>
ldap_port=<Port of LDAP server>
ldap_userdn=<DN of LDAP Admin User, usually "cn=orcladmin">
ldap_userpassword=<Password of LDAP Admin User>
oam_aaa_host=<HOST of OAM server>
oam_aaa_port=<Port of OAM server>
```

5. If your installation includes Content Server, then you also need to protect these URIs. Use the same parameters as the ones used in the previous steps so that the existing policy domain is updated with the URIs in the Content Server's `oam.conf` file.

```
java -jar ORACLE_HOME/modules/oracle.oamprovider_11.1.1/oamcfgtool.jar
mode=CREATE app_domain=<your_domain_name>
uris_file="WC_CONTENT_ORACLE_HOME/common/security/oam.conf"
app_agent_password=<Password to be provisioned for App Agent>
ldap_host=<Hostname of LDAP server>
ldap_port=<Port of LDAP server>
ldap_userdn=<DN of LDAP Admin User, usually "cn=orcladmin">
ldap_userpassword=<Password of LDAP Admin User>
oam_aaa_host=<HOST of OAM server>
oam_aaa_port=<Port of OAM server>
```

6. Check the Policy Domain settings.
   a. Log on to the Oracle Access Manager.
   b. Click **Policy Manager**.
   c. Click **My Policy Domains**.

      You should see the domain you just created in the list of policy domains. In the URL prefixes column, you should also see the URIs that were specified as part of the `webcenter.oam.conf` script file. You should also see the URIs from the SOA and Content Server OAM configuration files if you have run the `oamcfgtool` from SOA and Content Server domains.

   d. Click the domain you just created and open the Resources tab.

      The URIs you specified should display. You can also open other tabs to view and verify other settings, and manually add additional resources later, if required.

7. Check the Access Gate Configurations.
   a. Click **Access System Console**.
   b. Open the Access System Configuration tab.
   c. Click **AccessGate Configuration**.
   d. Enter some search criteria and click **Go**.
   e. When the Access Gate for the domain you just created displays (it will have the suffix `_AG`), click it to see the setting details.

8. Locate the policy domain that you created and verified in the previous steps and open the Policies tab.

   You should see four policies already created:

- WebCenter REST Policy
- Exclusion Scheme
- Protected_JSessionId_Policy
- Default Public Policy

9. Select WebCenter REST Policy, then select Authentication Rule and click Modify.
10. Change the AuthenticationScheme to OraDefaultBasicAuthNScheme (from OraDefaultFormAuthNScheme)
11. Go to Exclusion Scheme policy > Authentication Rule and check that it uses OraDefaultExclusionAuthNScheme (created in the previous steps) as the AuthenticationScheme.
12. Click **Save**.
13. Open the Policies tab and make sure that the polices are in the order shown below:

```
Protected_JSessionId_Policy
WebCenter REST Policy
Exclusion Scheme
Default Public Policy
```

14. Continue with the steps for installing the WebGate as described in Section 33.2.3.2.4, "Installing the WebGate 10g on the WebTier."

33.2.3.2.4 Installing the WebGate 10g on the WebTier

This section describes how to install the WebGate.

To install the WebGate:

1. Copy the ZIP file (Oracle_Access_Manager10_1_4_3_0_linux_GCClib.zip) containing the two gcc libraries required for the installation (libgcc_s.so.1 and libstdc++.so.5) to a /tmp directory. For more information, refer to the "Installing the WebGate" section in the *Oracle Access Manager Installation Guide*.
2. Run the installation as root. For example, from the /tmp directory run:

```
sudo -u root ./Oracle_Access_Manager10_1_4_3_0_linux_OHS11g_WebGate
```

3. Follow the installation runtime instructions, providing the installation directory, information of the AccessGate that you created earlier and the absolute path to the httpd.conf file of the web server. For example:

```
WT_ORACLE_HOME/instances/<your_instance>/config/OHS/ohs1/httpd.conf
```

   Information for the AccessGate can be found in the Access System Console.

4. After the installation, a new section is inserted in the httpd.conf file between the following entries:

```
#** BEGIN WEBGATE SPECIFIC ***
#** END Oblix NetPoint Specific ***
```

   Check to see if the content is consistent with your environment.

5. After installing and configuring the WebGate 10g, continue by configuring the Weblogic domain as described in Section 33.2.4, "Configuring the WebLogic Domain for OAM," and performing any additional service and component

configurations that apply as described in Section 33.2.6, "Additional Single Sign-on Configurations."

33.2.4 Configuring the WebLogic Domain for OAM

If your environment spans multiple domains (for example, a domain for WebCenter Portal, a separate domain for SOA, and a separate domain for Content Server), repeat the steps in this section for each domain.

This section includes the following subsections:

- Section 33.2.4.1, "Configuring the Oracle Internet Directory Authenticator"
- Section 33.2.4.2, "Configuring the OAM Identity Asserter"
- Section 33.2.4.3, "Configuring the Default Authenticator and Provider Order"
- Section 33.2.4.4, "Adding an OAM Single Sign-on Provider"

33.2.4.1 Configuring the Oracle Internet Directory Authenticator

Assuming Oracle Internet Directory is backing the OAM identity store, an Oracle Internet Directory authenticator (`OracleInternetDirectoryAuthenticator`) should be configured for the LDAP server that is used as the identity store of OAM, and the provider should be set to `SUFFICIENT`.

To configure the Oracle Internet Directory authenticator:

1. Log in to the WebLogic Server Administration Console.

   For information on logging in to the WebLogic Server Administration Console, see Section 1.13.2, "Oracle WebLogic Server Administration Console."

2. From the Domain Structure pane, click **Security Realms**.

   The Summary of Security Realms pane displays.

3. Click the realm entry for which to configure the OID authenticator.

   The Settings pane for the realm displays.

4. Open the Providers tab.

   The Provider Settings display.

5. Click **New** to create a provider.

   The Create a New Authentication Provider pane displays.

6. Enter a name for the new provider (for example, `OID Authenticator`), select `OracleInternetDirectoryAuthenticator` as its type and click **OK**.

7. On the Providers tab, click the newly added provider.

   The Common Settings pane for the authenticator displays.

8. Set the control flag to `SUFFICIENT` and click **Save**.

9. Open the Provider Specific tab.

   The Provider Specific Settings pane for the authenticator displays.

10. Complete the fields as shown in the table below. Leave the rest of the fields set to their default values.

Field	Value	Comment
Host:		The host ID for the LDAP server
Port:		The LDAP server port number
Principal:		The LDAP administrator principal (for example, cn=orcladmin)
Credential:	*<password>*	The administrator principal password
Confirm Credential:	*<password>*	
User Base DN:		User Search Base - this value should be the same as for the OAM Access Manager setup
All Users Filter:	"(&(uid=*)(objectclass =person))"	
User Name Attribute:	"uid"	
User From Name Filter:	"(&(uid=%u)(objectcl ass=person))"	The specified user name attribute must match in these three filters: "All Users Filter" and "User Name Attribute" and "User From Name Filter"
Group Base DN:		Group search base - Same as User Base DN
Use Retrieved User Name as Principal	Checked	User login IDs are usually case insensitive. This flag is required so that the subject established contains the user name as stored in the OID.

Note: The **User Name Attribute**, **All Users Filter**, and **Users From Name Filter** fields should all point to same OID attribute (`uid` in this case) and should match the Identity Store configuration for OAM.

11. Click **Save**.

33.2.4.2 Configuring the OAM Identity Asserter

An OAM identity asserter must be configured with the provider Control Flag set to `REQUIRED`.

To configure the OAM Identity asserter:

1. Log in to the WebLogic Server Administration Console.

   For information on logging in to the WebLogic Server Administration Console, see Section 1.13.2, "Oracle WebLogic Server Administration Console."

2. From the Domain Structure pane, click **Security Realms**.

   The Summary of Security Realms pane displays.

3. Click the realm entry for which to configure the OAM identity asserter.

   The Settings pane for the realm displays.

4. Open the Providers tab.

The Provider Settings display.

5. Click **New** to create a provider.

   The Create a New Authentication Provider pane displays.

6. Enter a name for the new provider (for example, `OAM ID Asserter`), select `OAMIdentityAsserter` as its type and click **OK**.

7. On the Providers tab, click the newly added provider.

   The Common Settings pane for the authenticator displays.

8. Set the control flag to `REQUIRED` and check that `OAM_REMOTE_USER` and `ObSSOCookie` is set for **Active Types**.

9. Click **Save** to save you settings.

33.2.4.3 Configuring the Default Authenticator and Provider Order

After configuring the OAM identity asserter, ensure that the default authenticator's control flag is set to `SUFFICIENT` and reorder the providers as shown below:

1. Navigate to the Provider Settings pane.
2. Open the Default Authenticator and check that the control flag is set to `SUFFICIENT`.
3. Do the same for any providers other than the two you just created.
4. On the Settings Pane, reset the provider order to:
   - `OAMIdentityAsserter` (`REQUIRED`)
   - `OracleInternetDirectoryAuthenticator` (`SUFFICIENT`)
   - `DefaultAuthenticator` (`SUFFICIENT`)
   - `DefaultIdentityAsserter`
5. Continue by configuring WebCenter Portal for single sign-on mode as described in Section 33.2.6.1, "Configuring WebCenter Portal for SSO." Also be sure to perform any further service and component configurations that apply to your environment as described in Section 33.2.6, "Additional Single Sign-on Configurations."

33.2.4.4 Adding an OAM Single Sign-on Provider

After checking that the default authenticator's control flag is set correctly, and that the order of the providers is correct, add an OAM SSO provider and restart all servers as described below.

> **Note:** This is required for OAM 11g, but is only required for OAM 10g if the logout URI has been explicitly configured.

1. Connect to the WebLogic domain using WLST and run the following command:

```
addOAMSSOProvider(loginuri="/${app.context}/adfAuthentication",
logouturi="/oamsso/logout.html")
```

2. Restart all servers.

33.2.5 Installing and Configuring the Oracle HTTP Server

This step is common to both OAM 10g and OAM 11g, and should be performed after installing and configuring OAM, and before configuring the WebLogic domain.

To install and configure the Oracle HTTP server (OHS).

1. If you do not have already have an OHS install you'd like to use, install the Oracle HTTP Server (11.1.1.4.0) using the instructions in the "Installing and Configuring Oracle Web Tier" section in *Oracle Fusion Middleware Installation Guide for Oracle Web Tier*. If you do have an existing installation, you will need to apply a patch to bring it up to OHS (11.1.1.4.0) as described in the "Applying the Latest Oracle Fusion Middleware Patch Set" section in the *Oracle Fusion Middleware Patching Guide*.
2. Configure WebTier OHS so that it forwards requests to the Oracle WebLogic Server for WebCenter Portal using the following example in `mod_wl_ohs.conf`. Make sure that the WebLogic port numbers match your configuration. For more information, see the "Installing and Configuring Oracle HTTP Server 11g Webgate for OAM" section in *Oracle Fusion Middleware Installing Webgates for Oracle Access Manager*.

> **Note:** This example assumes that WebCenter Portal is a non-cluster based installation. For a clustered environment change the `WebLogicHost` and `WebLogicPort` to `WeblogicCluster` as required for your environment. See the section on "Installing and Configuring Oracle HTTP Server 11g Webgate for OAM" in *Oracle Fusion Middleware Installing WebGates for Oracle Access Manager* for details.

```
# NOTE : This is a template to configure mod_weblogic.

LoadModule weblogic_module   "${ORACLE_HOME}/ohs/modules/mod_wl_ohs.so"

# This empty block is needed to save mod_wl related configuration from EM to
this file when changes are made at the Base Virtual Host Level
<IfModule weblogic_module>
#      WebLogicHost <WEBLOGIC_HOST>
#      WebLogicPort <WEBLOGIC_PORT>
#      Debug ON
#      WLLogFile /tmp/weblogic.log
#      MatchExpression *.jsp

<Location /webcenter>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8888
</Location>

<Location /webcenterhelp>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8888
</Location>

<Location /rss>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
```

```
      WebLogicPort 8888
</Location>

<Location /rest>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8888
</Location>

<Location /rsscrawl>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8888
</Location>

<Location /sesUserAuth>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8888
</Location>

<Location /owc_discussions>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8890
</Location>

<Location /activitygraph-engines>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8891
</Location>

<Location /wcps>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8891
</Location>

<Location /workflow>
      SetHandler weblogic-handler
      WebLogicHost soa.example.com
      WebLogicPort 8001
</Location>

<Location /integration/worklistapp>
      SetHandler weblogic-handler
      WebLogicHost soa.example.com
      WebLogicPort 8001
</Location>

<Location /integration/services>
      SetHandler weblogic-handler
      WebLogicHost soa.example.com
      WebLogicPort 8001
</Location>

<Location /soa-infra>
      SetHandler weblogic-handler
      WebLogicHost soa.example.com
```

```
      WebLogicPort 8001
</Location>

<Location /sdpmessaging/userprefs-ui>
      SetHandler weblogic-handler
      WebLogicHost soa.example.com
      WebLogicPort 8001
</Location>

<Location /DefaultToDoTaskFlow>
      SetHandler weblogic-handler
      WebLogicHost soa.example.com
      WebLogicPort 8001
</Location>

<Location /cs>
      SetHandler weblogic-handler
      WebLogicHost ucm.example.com
      WebLogicPort 16200
</Location>

<Location /adfAuthentication>
      SetHandler weblogic-handler
      WebLogicHost ucm.example.com
      WebLogicPort 16200
</Location>

<Location /pagelets>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8889
</Location>

<Location /services-producer>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8889
</Location>

<Location /wsrp-tools>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8889
</Location>

<Location /wcsdocs>
      SetHandler weblogic-handler
      WeblogicHost webcenter.example.com
      WeblogicPort 8888
</Location>

<Location /_vti_bin>
      SetHandler weblogic-handler
      WeblogicHost webcenter.example.com
      WeblogicPort 8888
</Location>

</IfModule>

# <Location /weblogic>
```

```
#      SetHandler weblogic-handler
#      PathTrim /weblogic
#      ErrorPage  http:/WEBLOGIC_HOME:WEBLOGIC_PORT/
#  </Location>
```

Note: The entries in the `Location` list above map the incoming paths to the appropriate WebLogic Server managed servers on which the corresponding applications reside.

33.2.6 Additional Single Sign-on Configurations

The configurations described in the following sections may be necessary or helpful in providing additional security for your site. After completing these configurations, continue by testing your OAM installation as described in Section 33.2.7, "Testing Your OAM Installation."

- Section 33.2.6.1, "Configuring WebCenter Portal for SSO"
- Section 33.2.6.2, "Configuring the Discussions Server for SSO"
- Section 33.2.6.3, "Configuring Worklists for SSO"
- Section 33.2.6.4, "Configuring OAM for RSS Feeds Using External Readers"
- Section 33.2.6.5, "Configuring the WebLogic Server Administration Console and Enterprise Manager for OAM 10g"
- Section 33.2.6.6, "Configuring the WebLogic Server Administration Console and Enterprise Manager for OAM 11g"
- Section 33.2.6.7, "Configuring Secure Enterprise Search for SSO"
- Section 33.2.6.8, "Configuring Content Server for SSO"
- Section 33.2.6.9, "Restricting Access with Connection Filters"
- Section 33.2.6.10, "Configuring Portlet Producers and Additional Components"

33.2.6.1 Configuring WebCenter Portal for SSO

Configure the WebCenter Portal application for SSO by adding a setting to `EXTRA_JAVA_PROPERTIES`.

There is a system property that tells WebCenter Portal and ADF that the application is configured in SSO mode and some special handling is required. The following system property is required in this mode:

Field	Value	Comment
`oracle.webcenter .spaces.osso`	`true`	This flag tells WebCenter Portal that SSO is being used, so no login form should be displayed on the default landing page. Instead, it displays a login link that the user can click to invoke the SSO authentication.

To set this property, edit the `setDomainEnv.sh` script located in your `<domain>/bin` directory, and add an entry like the following:

```
EXTRA_JAVA_PROPERTIES="-Doracle.webcenter.spaces.osso=true
${EXTRA_JAVA_PROPERTIES}"
export EXTRA_JAVA_PROPERTIES
```

After making this change, restart the `WC_Spaces` server.

33.2.6.2 Configuring the Discussions Server for SSO

This section describes how to configure the discussions server for single sign-on. Before configuring the discussions server for SSO, ensure that it has been configured to use the same identity store LDAP as WebCenter Portal, as described in Section 31.1, "Reassociating the Identity Store with an External LDAP Server." If you've chosen not to move the default administrator account to an external LDAP, be sure to also follow the instructions in Section 31.4.1, "Migrating the Discussions Server to Use an External LDAP."

> **Note:** Direct login to the discussions server is not supported after SSO is configured. Log in must be done through the Oracle HTTP Server URL.

To set up the discussions server for SSO:

1. Log in to the discussions server Admin Console at:

   `http://host:port/owc_discussions/admin`

   Where *host* and *port* are the host ID and port number of the `WC_Collaboration` managed server.

2. Open the System Properties page and edit (if it already exists) or add the `owc_discussions.sso.mode` property, setting it's value to `true`.

3. Edit or add the `jiveURL` property to point to the base URL of the WebTier. For example:

   ```
   jiveURL = webtier.example.com:7777/owc_discussions
   ```

   The `jiveURL` property is used when constructing links to forums in emails.

> **Note:**
>
> The registered WebCenter connection in WebCenter Portal for discussions and forums should point to the OHS URL.

33.2.6.2.1 Creating a Discussions Server Connection for WebCenter Portal

This section describes how to update the discussions server connection for WebCenter Portal so that it uses the WebTier's host and port values. Note that the steps below assume that the discussions component has already been installed and configured in the WebCenter Portal domain.

1. Using Fusion Middleware Control or WLST, change the Discussion server's URL host and port settings from the `WC_Spaces` managed server's settings, to the WebTier's host and port settings. For information about how to change these settings, see Section 12.5, "Modifying Discussions Server Connection Details."

2. Restart the `WC_Spaces` managed server.

   When you log in to WebCenter Portal, you automatically sign on to the Discussion server as well.

33.2.6.3 Configuring Worklists for SSO

Assuming that you've already set up a worklist connection, modify the URL to use the WebTier host and port instead of the SOA server host and port. You can do this using Fusion Middleware Control or using WLST commands as described in Section 20.4, "Setting Up Worklist Connections."

After modifying the URL and completing the setup required for OAM SSO, run the following command on the WebCenter Portal Administration server so that worklist changes take effect:

```
setBPELConnection('webcenter','WebCenter-Worklist',
'http://webtier.example.com:7777')
```

33.2.6.4 Configuring OAM for RSS Feeds Using External Readers

By default, WebCenter Portal RSS feeds are protected by SSO. However, they will not work well with external readers if left protected. If access using external readers is important, Oracle recommends that the WebCenter Portal RSS resource be excluded from the OAM policy so that the authentication for the RSS Servlet is handled by WebLogic Server's BASIC authentication that external readers can handle.

This section contains the following subsections:

- Section 33.2.6.4.1, "Unprotecting RSS Feeds in OAM 11g"
- Section 33.2.6.4.2, "Unprotecting RSS Feeds in OAM 10g"

33.2.6.4.1 Unprotecting RSS Feeds in OAM 11g

Follow the steps below to unprotect RSS feed for OAM 11g:

1. Open the OAM Admin Console.
2. Open the Policy Configuration tab and select **Application Domain > <your application domain>**.
3. Open the Resources tab and search for `/rss*`.

   Among the results, you should see:

   `/rss*`

   `/rss/.../*`

   `/rss/rssservlet*`

   `/rss/rssservlet/.../*`
4. For each resource, select the resource and click Edit.
5. Change each resource's Protection Level from `Protected` to `Excluded` and click Apply.

   Note that the resource's authentication policy and authorization policy are removed.
6. Close the tab and restart OHS.

33.2.6.4.2 Unprotecting RSS Feeds in OAM 10g

Follow the steps below to unprotect RSS feed for OAM 10g:

1. Open the OAM Admin Console.
2. Select **Access System Console > Policy Manager** and open the applicable policy domain.

3. Open the Policies tab, select the `Exclusion Scheme` policy, and click Modify.
4. Select the following resources for exclusion:

   `/rss`

   `/rss/rssservlet`
5. Click Save.
6. Select `Default Public Policy` and click Modify.
7. Uncheck the `/rss resource` and click Save.
8. Restart OHS.

33.2.6.5 Configuring the WebLogic Server Administration Console and Enterprise Manager for OAM 10g

This section describes how to optionally set up OAM single sign-on for the WebLogic Server Administration Console and Enterprise Manager.

> **Note:** Setting up OAM SSO for Enterprise Manager and the WebLogic Server Administration Console would provide single sign-on access to same set of users for whom OAM SSO access has been configured. If want the WebTier to be accessible to external users through OAM, but want administrators to log in directly to Enterprise Manager and the WebLogic Server Administration Console, then you may not want to complete this additional configuration step.

To set up OAM SSO for the WebLogic Server Administration Console and Enterprise Manager:

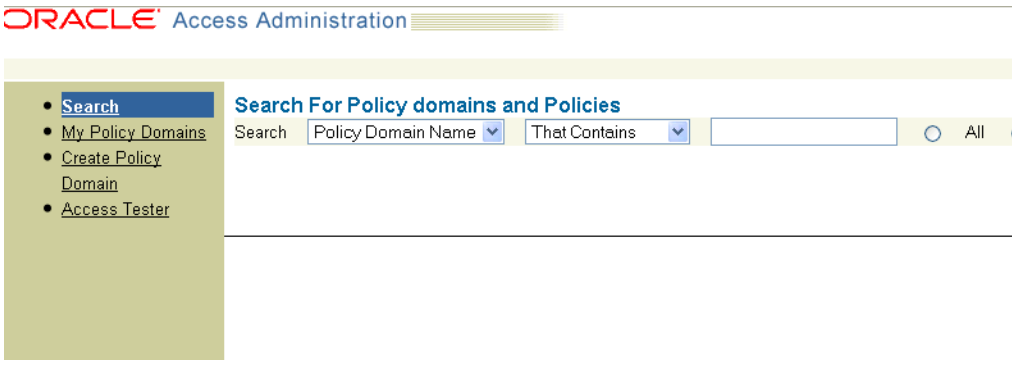
1. Log in to the OAM Console using your browser to navigate to:

   `http://`*`host`*`:`*`port`*`/access/oblix`

   Where *`host`* is the host ID of the server hosting the Access Manager (for example, `oam.example.com`), and *`port`* is the HTTP port number (for example, `8888`).
2. Click **Policy Manager**.

   The Policy Manager pane displays (see Figure 33–4).

Figure 33–4 Policy Manager Pane

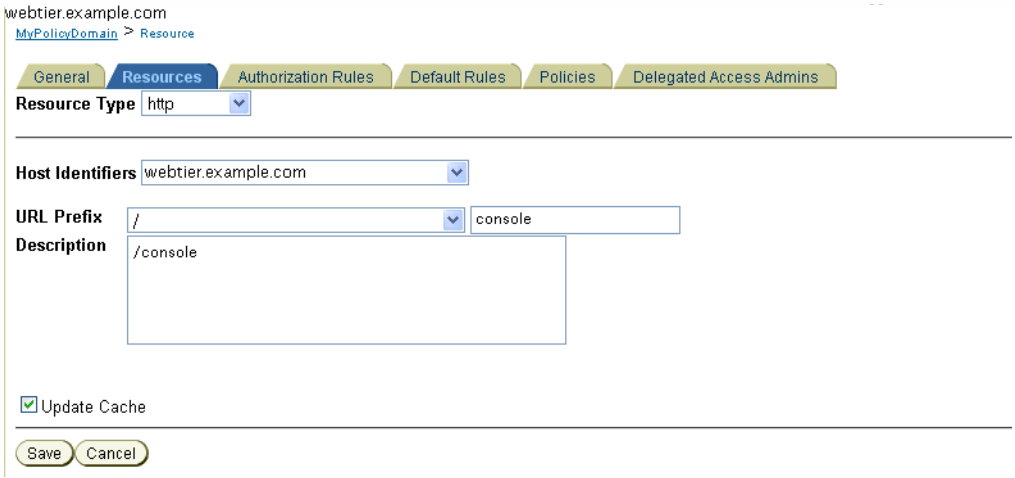


3. Locate the policy domain that you created to protect WebCenter Portal resources.

4. Open the Resources tab and click **Add**.

   The Resource page displays (see Figure 33–5).

Figure 33–5 Policy Domain Resource Page



5. Add the resources that must be secured. For each resource:

   a. Select `http` as the **Resource Type**.

   b. Select the **Host Identifier** for the WebCenter Portal WebTier.

   c. Enter the **URL Prefix** for the WebLogic Server Administration Console (`/console`) or Enterprise Manager (`/em`).

   d. Enter a **Description** for the resource.

   e. Ensure that **Update Cache** is selected, and then click **Save**.

6. In your WebTier, modify the `mod_wl_ohs.conf` file (in `WT_ORACLE_HOME/instances/<your_instance>/config/OHS/ohs1/`) to include the WebLogic Server Administration Console and Enterprise Manager, by adding two additional Location entries using the actual host ID for the WebCenter Portal Administration Server for `WebLogicHost`.

```
<Location /console>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 7001
</Location>

<Location /em>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 7001
</Location>
```

7. Restart the Oracle HTTP Server for your changes to take effect.

   You should now be able to access the WebLogic Server Administration Console and Enterprise Manager with the following links:

```
http://host:OHS port/console
http://host:OHS port/em
```

and be prompted with the OAM SSO login form.

33.2.6.6 Configuring the WebLogic Server Administration Console and Enterprise Manager for OAM 11g

This section describes how to optionally set up OAM 11g single sign-on for the WebLogic Server Administration Console and Enterprise Manager.

> **Note:** Setting up OAM SSO for Enterprise Manager and the WebLogic Server Administration Console would provide single sign-on access to same set of users for whom OAM SSO access has been configured. If want the WebTier to be accessible to external users through OAM, but want administrators to log in directly to Enterprise Manager and the WebLogic Server Administration Console, then you may not want to complete this additional configuration step.

To set up OAM 11g SSO for the WebLogic Server Administration Console and Enterprise Manager:

1. Log in to the OAM Console using your browser:

   `http://host:port/oamconsole`

2. Go to **Policy Configuration > Application Domains**.

   The Policy Manager pane displays.

3. Locate the application domain you created using the name while registering webgate agent.

4. Expand the Resources node and click **Create**.

   The Resource page displays.

5. Add the resources that must be secured. For each resource:

   a. Select `http` as the **Resource Type**.

   b. Select the **Host Identifier** created while registering the WebGate agent.

   c. Enter the **Resource URL** for the WebLogic Server Administration Console (`/console /console/* /console/.../*`) or Enterprise Manager (`/em /em/* /em/.../*`).

   d. Enter a **Description** for the resource and click **Apply**.

6. Go to **Authentication Policies > Protected Resource Policy** and add the newly created resource.

7. Do the same under **Authorization Policies > Protected Resource Policy**>

8. In your WebTier, modify the `mod_wl_ohs.conf` file (in `WT_ORACLE_HOME/instances/<your_instance>/config/OHS/ohs1/`) to include the WebLogic Server Administration Console and Enterprise Manager, by adding two additional Location entries using the actual host ID for the WebCenter Portal Administration Server for `WebLogicHost`.

```
<Location /console>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 7001
</Location>
```

```
<Location /em>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 7001
</Location>
```

9. Restart the Oracle HTTP Server for your changes to take effect.

   You should now be able to access the WebLogic Server Administration Console and Enterprise Manager with the following links:

   ```
   http://host:OHS port/console
   http://host:OHS port/em
   ```

   and be prompted with the OAM SSO login form.

33.2.6.7 Configuring Secure Enterprise Search for SSO

The crawl sources that are defined to crawl WebCenter Portal data and repositories used by WebCenter Portal and the corresponding authentication end points defined in SES must be routed through the WebTier OHS ports so that they can be properly authenticated (the authentication method continues to be BASIC and realm jazn.com). For information about configuring SES connections, see Section 18.4, "Setting Up Oracle SES Connections."

33.2.6.8 Configuring Content Server for SSO

After you've completed your SSO setup, and after setting up a connection for Content Server, specify the web context root in the `JCRContentServerConnection` using Fusion Middleware Control, or as shown in the following WLST example:

```
setJCRContentServerConnection(appName, name, webContextRoot='/cs')
```

Setting the web context root tells the Document Library code that SSO has been set up. Note that this setting should *not* be set until after SSO has been completely set up.

33.2.6.9 Restricting Access with Connection Filters

Follow the steps below to only allow users to access WebCenter Portal or Portal Framework applications and associated components through the WebTier OHS ports so that they can be properly authenticated.

1. Log in to the WebLogic Server Administration Console.

   For information on logging in to the WebLogic Server Administration Console, see Section 1.13.2, "Oracle WebLogic Server Administration Console."

2. In the Domain Structure pane, select the domain you want to configure (for example, `webcenter`).

3. Open the **Security** tab and the **Filter** subtab.

   The Security Filter Settings pane displays.

4. Check **Connection Logger Enabled** to enable the logging of accepted messages.

   The Connection Logger logs successful connections and connection data in the server. You can use this information to debug problems relating to server connections.

5. In the **Connection Filter** field, specify the connection filter class to be used in the domain.

- To configure the default connection filter, specify `weblogic.security.net.ConnectionFilterImpl`.
- To configure a custom connection filter, specify the class that implements the network connection filter. Note that this class must also be present in the CLASSPATH for WebLogic Server.

6. In the Connection Filter Rules field, enter the syntax for the connection filter rules.

   For example:

   ```
   <webtier IP>/0 * * allow
   0.0.0.0/0  *  *  deny
   ```

   which says: allow all traffic coming from the local host and disallow all traffic from any other IP address. You should, of course, write the network filter(s) that are relevant to your environment. For more information about writing connection filters, see the "Developing Custom Connection Filters" section in *Oracle Fusion Middleware Programming Security for Oracle WebLogic Server*.

7. Click **Save** and activate the changes.
8. Restart all the managed servers and the Administration server.
9. Verify that all direct traffic to the WebLogic Server is blocked by attempting to navigate to:

   `http://host:WLS_port/webcenter`

   This should produce the following error:

   ```
   "The Server is not able to service this request:
   [Socket:000445]Connection rejected, filter blocked Socket,
   weblogic.security.net.FilterException: [Security:090220]rule
   3"
   ```

   You should, however, still be able to access WebCenter Portal through the OHS port:

   `http://host:OHS_port/webcenter`

33.2.6.10 Configuring Portlet Producers and Additional Components

If you have set up your Portlet Producer applications to route through OHS, be sure to use the OHS host and port when specifying producer URLs for registration. This applies to out of-the-box producers like wsrp-tools, services-producer, pagelet producers and any other producer you have explicitly configured.

33.2.7 Testing Your OAM Installation

After installing and configuring either OAM 10g or 11g, check that you can access all of the configured applications below (as they apply to your environment), and that the global login and logout is giving you access to all of your configured applications without prompting you to sign in again. Also test global logout where available and make sure you are logged out of all other related applications.

- **WebCenter Portal:** Access any protected WebCenter Portal URL (a protected portal, for example), and make sure that you see the SSO login challenge. If you are already logged into another related application that uses the same SSO, you should automatically be shown content.
- **REST:** Access `http://ohshost:ohsport/rest/api/resourceIndex`. You should see the BASIC authentication challenge. If you are already logged into

another related application that uses the same SSO, you should automatically be shown content.

- **REST:** Access `http://ohshost:ohsport/rest/api/cmis/....` (retrieve this from `resourceIndex` access output in the previous step). You should not see a login challenge and should be able to see public content. When you access this after you've logged in, then you should see all content to which you have access rights.
- **ActivityGraph Engines:** Access `http://host:port/activitygraph-engines`. You should see an SSO login challenge. Once logged in, you should be able to see content.
- **Content Server**: Go to the profile UI and check that you can see Content Server screens embedded in iFrames without challenging you to log in. You should also be able to access Site Studio content in Content Presenter templates without logging in as you are already logged into WebCenter Portal.
- **SOA:** Access links in a workflow task flow and make sure that you are not challenged to log in.
- **Discussion forums:** Access the discussions application at `http://host:port/owc_discussions` and log in. Check that the login is the SSO login challenge. Similarly, the Administration login to the discussions server at `http://host:port/owc_discussions/admin` should also go through the SSO login challenge.

33.3 Configuring Oracle Single Sign-On (OSSO)

In a default installation, WebCenter Portal and Portal Framework applications use the HTTP ports in the managed server created for them. To configure WebCenter Portal and Portal Framework applications with Oracle Single Sign-On, the application needs Oracle HTTP Server and the associated Module `mod_osso` to integrate with Oracle Single Sign-On (OSSO). Note that for Portal Framework applications some additional configurations are required, as described in Section 34.5, "Configuring Portal Framework Applications for OSSO."

> **Note:** The BPEL Console does not support SSO integration. When WebCenter Portal or a Portal Framework application is configured for SSO, login to BPEL must still be done through the standard login page on the BPEL Console.

This section includes the following subsections

33.3.1 Roadmap to Configuring OSSO

The flow chart (Figure 33–6) and table (Table 33–2) in this section provide an overview of the prerequisites and tasks required to configure single sign-on for WebCenter Portal and Portal Framework applications using OSSO.

Figure 33–6 Configuring Single Sign-on Using OSSO

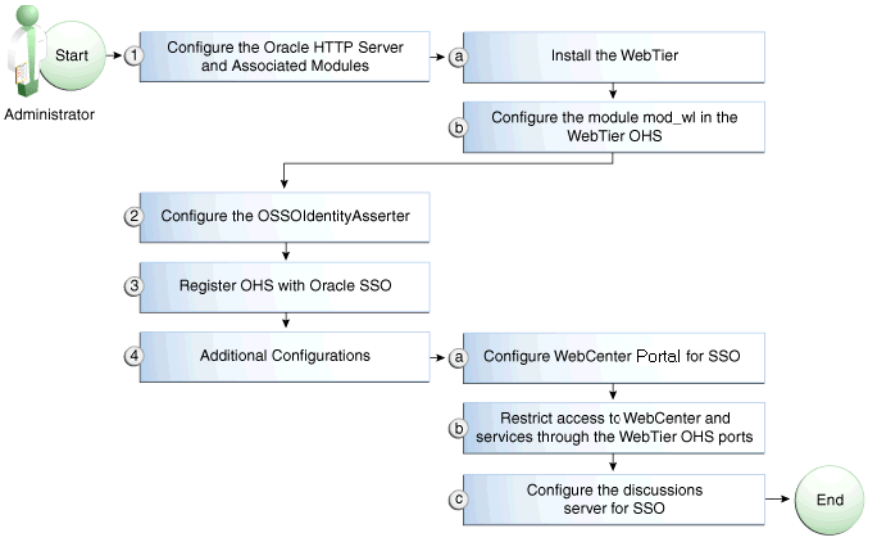


Table 33–2 shows the tasks and sub-tasks for configuring single sign-on for WebCenter Portal using OSSO.

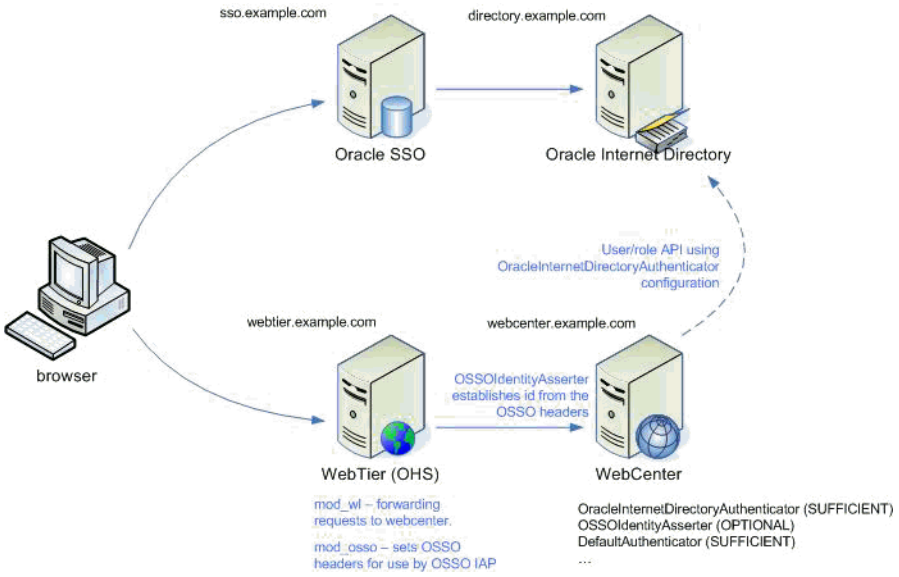
Table 33–2 Configuring Single Sign-on for WebCenter Portal Using OSSO

Actor	Task	Sub-task	Notes
Administrator	**1.** Configure the Oracle HTTP Server and Associated Modules	**1.a** Install the WebTier	
		1.b Configure the module mod_wl in the WebTier OHS	
	2. Configure the OSSOIdentityAsserter		
	3. Register OHS with Oracle SSO		
	4. Perform additional configurations as required	**4.a** Configure WebCenter Portal for SSO	
		4.b Restrict access to WebCenter and components and services through the WebTier OHS ports	
		4.c Configure the discussions server for SSO	

33.3.2 OSSO Components and Topology

In a default installation, WebCenter Portal and Portal Framework applications use the HTTP ports of the managed server created for them. To configure WebCenter Portal or Portal Framework application with Oracle Single Sign-On, the application needs the Oracle HTTP Server and the associated Module `mod_osso`, to integrate with Oracle SSO. The diagram below (Figure 33–7) shows the overall architecture of this integration:

Figure 33–7 OSSO Components and Topology



33.3.3 Configuring the Oracle HTTP Server and Associated Modules

This section describes how to load and configure the Oracle HTTP Server and associated modules.

To load and configure the Oracle HTTP Server and associated mods:

1. Install the Oracle WebTier software, which contains Oracle HTTP Server (OHS) and associated mods (`mod_osso` and `mod_wl`).
2. Configure the module `mod_wl` in WebTier OHS so that it forwards requests to the Oracle WebLogic Server for WebCenter Portal or Portal Framework application, replacing the host and port values with those for you local environment.

   Uncomment the lines at `${ORACLE_HOME}/ohs/modules/mod_wl_ohs.so`. This file is included by the `httpd.conf` file and looks like the following:

```
# NOTE : This is a template to configure mod_weblogic.

LoadModule weblogic_module   "${ORACLE_HOME}/ohs/modules/mod_wl_ohs.so"

# This empty block is needed to save mod_wl related configuration from EM to
this file when changes are made at the Base Virtual Host Level
<IfModule weblogic_module>
#      WebLogicHost <WEBLOGIC_HOST>
#      WebLogicPort <WEBLOGIC_PORT>
#      Debug ON
#      WLLogFile /tmp/weblogic.log
#      MatchExpression *.jsp

<Location /webcenter>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8888
</Location>

<Location /webcenterhelp>
      SetHandler weblogic-handler
```

```
      WebLogicHost webcenter.example.com
      WebLogicPort 8888
</Location>

<Location /rss>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8888
</Location>

<Location /rest>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8888
</Location>

<Location /rsscrawl>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8888
</Location>

<Location /sesUserAuth>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8888
</Location>

<Location /services-producer>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8889
</Location>

<Location /wsrp-tools>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8889
</Location>

<Location /owc_discussions>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8890
</Location>

<Location /activitygraph-engines>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8891
</Location>

<Location /wcps>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8891
</Location>

<Location /workflow>
      SetHandler weblogic-handler
```

```
      WebLogicHost soa.example.com
      WebLogicPort 8001
</Location>

<Location /integration/worklistapp>
      SetHandler weblogic-handler
      WebLogicHost soa.example.com
      WebLogicPort 8001
</Location>

<Location /integration/services>
      SetHandler weblogic-handler
      WebLogicHost soa.example.com
      WebLogicPort 8001
</Location>

<Location /soa-infra>
      SetHandler weblogic-handler
      WebLogicHost soa.example.com
      WebLogicPort 8001
</Location>

<Location /sdpmessaging/userprefs-ui>
      SetHandler weblogic-handler
      WebLogicHost soa.example.com
      WebLogicPort 8001
</Location>

<Location /DefaultToDoTaskFlow>
      SetHandler weblogic-handler
      WebLogicHost soa.example.com
      WebLogicPort 8001
</Location>

<Location /cs>
      SetHandler weblogic-handler
      WebLogicHost ucm.example.com
      WebLogicPort 16200
</Location>

<Location /adfAuthentication>
      SetHandler weblogic-handler
      WebLogicHost ucm.example.com
      WebLogicPort 16200
</Location>

</IfModule>


# <Location /weblogic>
#      SetHandler weblogic-handler
#      PathTrim /weblogic
#      ErrorPage  http:/WEBLOGIC_HOME:WEBLOGIC_PORT/
#  </Location>
```

33.3.4 Configuring the OSSOIdentityAsserter

Include the OSSO Identity Assertion Provider (IAP) provider in the Oracle WebLogic domain for WebCenter Portal or Portal Framework application. Use the WebLogic

Server Administration Console to add the OSSO IAP to your domain as shown in the steps below. If your environment spans multiple domains (for example, a domain for WebCenter Portal, separate domain for SOA, and a separate domain for Content Server), repeat the steps in this section for each domain.

To configure the OSSOIdentityAsserter:

1. Log in to the WebLogic Server Administration Console.

   For information on logging in to the WebLogic Server Administration Console, see Section 1.13.2, "Oracle WebLogic Server Administration Console."

2. From the Domain Structure pane, click **Security Realms**.

   The Summary of Security Realms pane displays.

3. Click the realm entry to which to add the provider.

   The Settings pane for the realm displays.

4. Click the Providers tab.

   The Provider Settings display.

5. Click **New** to create a provider.

   The Create a New Authentication Provider pane displays.

6. Enter a name for the new provider, select **OSSOIdentityAsserter** as its type and click **OK**.

7. On the Providers tab, click the newly added provider.

8. Set the control flag to `OPTIONAL`.

9. Ensure that **OracleInternetDirectoryAuthenticator** (or the primary authenticator you selected when you configured the Identity Store to use an external LDAP) is set as the primary authenticator for the domain so that the user profile can be retrieved from the associated Oracle Internet Directory server. For information about configuring the Identity Store to use an external LDAP, see Chapter 31, "Configuring the Identity Store."

   For OID, the provider list should appear as follows:

   - **OSSOIdentityAsserter** (`OPTIONAL`)
   - **OracleInternetDirectoryAuthenticator** (`SUFFICIENT`)
   - **DefaultAuthenticator** (`SUFFICIENT`)
   - **DefaultIdentityAsserter** (`OPTIONAL`)

33.3.5 Registering OHS with Oracle SSO

Register the module `mod_osso` in the WebTier OHS with the SSO Server as a partner application by following the steps below.

To register OHS with Oracle SSO:

1. Run `ssoreg` from the SSO server to generate an `osso.conf` file and FTP it in binary mode to the WebTier host (*`WT_ORACLE_HOME`*).

   The following example shows how you would register a remote partner application on a SSO Server. Check that the *`ORACLE_HOME`* environment variable is set (*`ORACLE_HOME`* here is the *`ORACLE_HOME`* of the OSSO installation on the SSO server) before running `ssoreg.sh`.

```
bash-3.00$ ORACLE_HOME/sso/bin/ssoreg.sh -site_name
webtier.example.com:80 -config_mod_osso TRUE -mod_osso_url
http://webtier.example.com:80 -remote_midtier  -config_file
webtier.example.com.osso.conf
```

Running this command creates a `webtier.example.com.osso.conf` file.

2. Copy the `WT_ORACLE_HOME/instances/<your_instance>/config/OHS/ohs1/disabled/mod_osso.conf` file to `WT_ORACLE_HOME/instances/<your_instance>/config/OHS/ohs1/moduleconf`. All files in the `moduleconf` directory are included in the `httpd.conf` file.
3. Copy the `webtier.example.com.osso.conf` file generated by `ssoreg` in **step 1** to a location accessible in the WebTier other than the `moduleconf` directory (for example, WT_ORACLE_HOME).

> **Note:** If using FTP, be sure to transfer the file using binary mode.

4. Add rules to the `mod_osso.conf` file to protect the `/webcenter` and related application resources URLs with Oracle SSO.

   The `mod_osso.conf` file should look similar to this:

```
LoadModule osso_module "${ORACLE_HOME}/ohs/modules/mod_osso.so"

<IfModule osso_module>
    OssoIpCheck off
    OssoIdleTimeout off
    OssoSecureCookies Off

#Point to proper osso.conf file.
   OssoConfigFile /example.com.osso.conf
#
# Insert Protected Resources: (see Notes below for
# how to protect resources)
#
#______-
#
# Notes
#
#______-
#
# 1. Here's what you need to add to protect a resource,
#    e.g. <ApacheServerRoot>/htdocs/private:
#
#      <Location /private>
#      require valid-user
#      AuthType Osso
#      </Location>

    <Location /webcenter/adfAuthentication*>
      OssoSendCacheHeaders off
      require valid-user
      AuthType Osso
    </Location>
    <Location /services-producer/adfAuthentication*>
       OssoSendCacheHeaders off
```

```
   require valid-user
   AuthType Osso
</Location>
<Location /rss/rssservlet>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
<Location /owc_discussions/login!withRedirect.jspa>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
<Location /owc_discussions/login!default.jspa>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
<Location /owc_discussions/login.jspa>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
<Location /owc_discussions/admin>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
<Location /integration/worklistapp>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
<Location /sdpmessaging/userprefs-ui>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
<Location /workflow/WebCenterWorklistDetail>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
<Location /workflow/sdpmessagingsca-ui-worklist>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
<Location /rest/api/resourceIndex>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
<Location /rest/api/spaces>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
<Location /rest/api/discussions>
  OssoSendCacheHeaders off
```

```
  require valid-user
  AuthType Osso
</Location>
<Location /rest/api/tags>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
<Location /rest/api/taggeditems>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
<Location /rest/api/activities>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
<Location /rest/api/activitygraph>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
<Location /rest/api/feedback>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
<Location /rest/api/people>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
<Location /rest/api/messageBoards>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
<Location /rest/api/searchresults>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
<Location /pagelets/admin>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
<Location /pagelets/authenticateWithApplicationServer*>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
<Location /activitygraph-engines>
  OssoSendCacheHeaders off
  require activity-graph-admins
  AuthType Osso
</Location>
<Location /wcps/api>
  OssoSendCacheHeaders off
```

```
      require valid-user
      AuthType Osso
    </Location>
    <Location /cs/groups>
       OssoSendCacheHeaders off
       require valid-user
       AuthType Osso
    </Location>
    <Location /cs/idcplg>
       OssoSendCacheHeaders off
       require valid-user
       AuthType Osso
    </Location>
   <Location /adfAuthentication>
      OssoSendCacheHeaders off
      require valid-user
      AuthType Osso
   </Location>
</IfModule>

#
# To have short hostnames redirected to fully qualified
# hostnames for clients that need authentication via
# mod_osso to be able to enter short hostnames into their
# browsers use a mod_rewrite configuration such as the following.
#
# e.g
#RewriteEngine On
#RewriteCond %{HTTP_HOST} !www.example.com
#RewriteRule ^.*$
http://%{SERVER_NAME}%{REQUEST_URI}
[R]
#where www.example.com is the fully qualified domain name.
```

Be sure to change the **`OssoConfigFile`** parameter to point to the location (and filename if you've changed it) where you copied your `osso.conf` file in the previous step. If your environment is non-SSL, then also be sure to turn off OSSO secure cookies (on by default):

```
OssoSecureCookies Off
```

5. Restart the WebTier so that the configuration changes to `mod_osso` and `mod_wl` take effect.

33.3.6 Additional Configurations

The configurations described in the following sections may be necessary or helpful in providing additional security for your site. For Portal Framework applications the additional configurations described in Section 34.5, "Configuring Portal Framework Applications for OSSO" are also required.

- Section 33.3.6.1, "Configuring WebCenter Portal for SSO"
- Section 33.3.6.2, "Restricting Access Using the WebTier OHS Ports"
- Section 33.3.6.3, "Configuring the Discussions Server for SSO"
- Section 33.3.6.4, "Configuring the Worklist Component for SSO"
- Section 33.3.6.5, "Configuring Oracle Content Server for SSO"
- Section 33.3.6.6, "Configuring OSSO for RSS Feeds Using External Readers"

- Section 33.3.6.7, "Configuring SES Crawl for SSO"

33.3.6.1 Configuring WebCenter Portal for SSO

Complete the configuration for Oracle Single Sign-on (OSSO) for WebCenter Portal by adding a setting to `EXTRA_JAVA_PROPERTIES` and rebooting as described in Section 33.2.6.1, "Configuring WebCenter Portal for SSO."

33.3.6.2 Restricting Access Using the WebTier OHS Ports

To only allow users to access WebCenter Portal or Portal Framework application and associated components and services through the WebTier OHS ports so that they can be properly authenticated, follow the steps in Section 33.2.6.9, "Restricting Access with Connection Filters."

33.3.6.3 Configuring the Discussions Server for SSO

This section describes how to configure the discussions server for single sign-on. Before configuring the discussions server for SSO, ensure that it has been configured to use the same identity store LDAP as WebCenter Portal, as described in Section 31.4.1, "Migrating the Discussions Server to Use an External LDAP."

To set up the discussions server for SSO:

1. Log in to the discussions server Admin Console at:

   `http://`*`host`*`:`*`port`*`/owc_discussions/admin`

   Where *`host`* and *`port`* are the host ID and port number of the `WC_Collaboration` managed server.

2. Open the System Properties page and edit (if it already exists) or add the `owc_discussions.sso.mode` property, setting it's value to `true`.

3. Update the `jiveURL` property to point to the base URL of the WebTier.

33.3.6.4 Configuring the Worklist Component for SSO

After registering OHS with Oracle SSO, as shown in Section 33.3.5, "Registering OHS with Oracle SSO," run the following command using the WebCenter Portal Administration server so that the worklist changes to take effect:

```
setBPELConnection('webcenter','WebCenter-Worklist',
'http://webtier.example.com:7777')
```

33.3.6.5 Configuring Oracle Content Server for SSO

Since it's possible to access the Content Server repository directly from WebCenter Portal, you may also want to include it in the single sign-on configuration. Assuming that you've already set up a connection for the Content Server, specify the web context root in the `JCRContentServerConnection` using Fusion Middleware Control or using WLST as shown in the following example:

```
setJCRContentServerConnection(appName, name, webContextRoot='/cs')
```

For more information on how to configure the Content Server, see the "Configuring WebCenter Content for Single Sign-On" section in *Oracle Fusion Middleware System Administrator's Guide for Oracle Content Server*.

33.3.6.6 Configuring OSSO for RSS Feeds Using External Readers

By default, WebCenter Portal RSS feeds are protected by SSO. However, they will not work well with external readers if left protected. If access using external readers is important, Oracle recommends that the WebCenter Portal RSS resource be unprotected so that the authentication for the RSS Servlet is handled by WebLogic Server's BASIC authentication that external readers can handle.

Follow the steps below to unprotect the RSS feeds:

1. Remove the following entry from `mod_osso.conf`.

```
<Location /rss/rssservlet>
  OssoSendCacheHeaders off
  require valid-user
  AuthType Osso
</Location>
```

2. Restart OHS.

33.3.6.7 Configuring SES Crawl for SSO

If you have SES configured for your instance, you can optionally update the WebCenter Portal Crawl and authentication end points to use the WebTier URL. See Chapter 18, "Managing Oracle Secure Enterprise Search in WebCenter Portal" for more information.

33.4 Configuring SAML-based Single Sign-on

Security Assertion Markup Language (SAML) enables cross-platform authentication between Web applications or Web services running in a WebLogic Server domain and Web browsers or other HTTP clients. WebLogic Server supports single sign-on (SSO) based on SAML for WebCenter Portal and Portal Framework applications (Pagelet Producer applications are not supported). When users are authenticated at one site that participates in a single sign-on (SSO) configuration, they are automatically authenticated at other sites in the SSO configuration and do not need to log in separately. Note that since Pagelet Producer applications do not participate in SAML SSO, users are required to log in explicitly if they accesses the Pagelet Producer application. Note also that for Portal Framework applications, some additional configurations are required as described in Section 34.6, "Configuring Portal Framework Applications for SAML SSO."

Note: Although SAML-based single sign-on provides support for logging users onto subsequent applications after initial sign-on, global logout is not supported. Consequently, users must log out of each individual application they open.

Note also that if you set up SAML-based single sign-on with WebCenter Portal as the source application and discussions as the destination application, administrators can access the discussions administration pages from WebCenter Portal Administration (**Configuration > Services**) and Portal Settings (Services page). However, since discussions administration pages do not participate in SSO, if you access administration pages directly, you are required to log in to the discussions server again.

Finally, SAML-based single sign-on is not available for the `sdpmessaging userprefs-ui` application. As an application administrator, if you click **Manage Configuration** in the **Preferences > Messaging** dialog in WebCenter Portal, you will need to log in again.

This SSO mechanism can be used for departmental installations for which there is no existing Oracle SSO or Oracle Access Manager single sign-on infrastructure, but single sign-on between only WebCenter Portal and its components or services is required. For High Availability and large enterprise deployments, Oracle Access Manager SSO is recommended.

This section describes how to set up SAML 1.1-based single sign-on for WebCenter Portal or Portal Framework application and worklists running on different managed servers within the same domain.

This section includes the following subsections:

- Section 33.4.1, "SAML Components and Topology"
- Section 33.4.2, "Configuring SAML-based Single Sign-on"

33.4.1 SAML Components and Topology

Figure 33–9 shows the components and their interaction in a SAML-based single sign-on configuration that includes WebCenter Portal and discussions.

A SAML-based single sign-on solution consists of the following components:

- **SAML Credential Mapper** - The SAML Credential Mapping provider acts as a producer of SAML security assertions, allowing WebLogic Server to act as a source site for using SAML for single sign-on. The SAML Credential Mapping provider generates valid SAML 1.1 assertions for authenticated subjects based on the configuration of the target site or resource.
- **Inter Site Transfer Service (ITS)** - an addressable component that generates identity assertions and transfers the user to the destination site.
- **Assertion Retrieval Service (ARS)** - an addressable component that returns the SAML assertion that corresponds to the artifact. The assertion ID must have been allocated at the time assertion was generated.
- **SAML Identity Asserter** - The SAML Identity Assertion provider acts as a consumer of SAML security assertions, allowing WebLogic Server to act as a destination site for using SAML for single sign-on. The SAML Identity Assertion

provider processes valid SAML 1.1 assertions for authenticated subjects obtained from the source site or resource.

- **Assertion Consumer Service (ACS)** - an addressable component that receives assertions and/or artifacts generated by the ITS and uses them to authenticate users at the destination site
- **SAML Relying party** - A SAML Relying Party is an entity that relies on the information in a SAML assertion produced by the SAML source site. You can configure how WebLogic Server produces SAML assertions separately for each Relying Party or use the defaults established by the Federation Services source site configuration for producing assertion.
- **SAML Asserting party** - A SAML Asserting Party is a trusted SAML Authority (an entity that can authoritatively assert security information in the form of SAML Assertions).

Figure 33–8 shows the components and flow for a POST-configured SAML SSO configuration that includes both a WebCenter Portal and SOA domain. The flow is similar for other destination applications participating in single sign-on such as Worklist applications and discussions.

Figure 33–8 Detailed SAML Single Sign-on Components and Topology (POST Profile Configured)

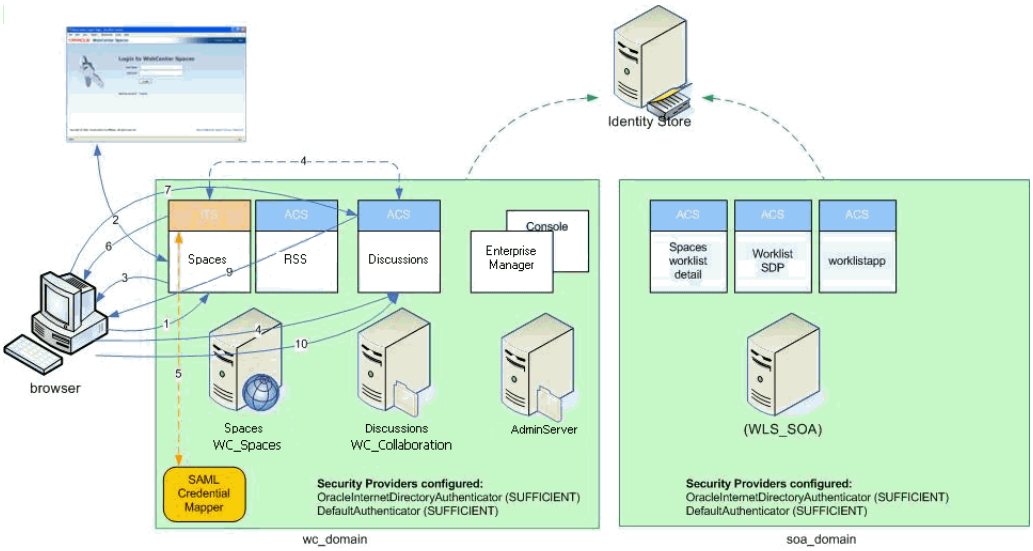
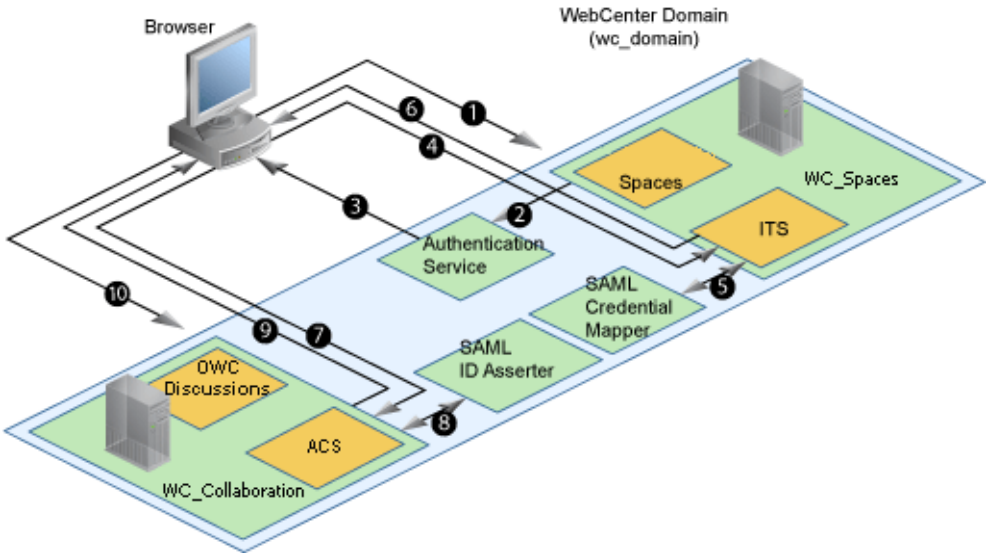


Figure 33–9 shows a simplified version of the components and flow for a POST-configured SAML SSO configuration, including the SAML SSO flow between WebCenter Portal and the discussions application.

Figure 33–9 SAML Single Sign-on Components and Topology (POST Profile Configured)



The steps in the flow are:

1. The user's browser accesses WebCenter Portal (source site), hosted on a WebLogic managed server (`WC_Spaces`) in the WebCenter Portal domain (`wc_domain`), by supplying user credentials.
2. WebCenter Portal passes the user credentials to the authentication service provider.
3. If authentication is successful, the authenticated session is established, and the WebCenter Portal welcome page is displayed.
4. From the welcome page, the user then clicks on a link on the page to access a secured Web page of the discussions destination site, hosted on a different WebLogic Server (`WC_Collaboration`) in the same domain. This triggers a call to the Inter-Site Transfer Service (ITS) servlet configured. In this case, the ITS servlet is hosted within the source site (that is, on the WebCenter Portal application on the `WC_Spaces` managed server) that shares the same JSESSIONID cookie as WebCenter Portal.
5. The ITS servlet calls the SAML Credential Mapper configured in the WebCenter Portal domain (`wc_domain`) to request a caller assertion. The SAML Credential Mapper returns the assertion. It also returns the URL of the destination site application Web page (a secured Web page for discussions) and path to the appropriate POST form (if the source site is configured to use the POST profile).
6. The SAML ITS servlet generates a SAML response containing the generated assertion, signs it, base-64 encodes it, embeds it in the HTML form, and returns the form to the user's browser.
7. The user's browser POSTs the form to the destination site's Assertion Consumer Service (ACS). In this case, the ACS Servlet is hosted in destination site (discussions) and shares its login cookie.
8. The assertion is validated.
9. If the assertion is successful, the user is redirected to the target (the secured Web page for discussions).

10. The user is logged in on the destination site (discussions) without having to reauthenticate.

33.4.2 Configuring SAML-based Single Sign-on

This section describes how to configure WebCenter Portal and associated services and components for SAML-based single sign-on using a set of automated scripts.

This section includes the following subsections:

- Section 33.4.2.1, "SAML Single Sign-on Prerequisites"
- Section 33.4.2.2, "Configuring SAML-based SSO"
- Section 33.4.2.3, "Configuring SAML SSO for RSS Using External Readers"
- Section 33.4.2.4, "Checking Your Configuration"
- Section 33.4.2.5, "Disabling Your SAML SSO Configuration"
- Section 33.4.2.6, "Removing Your SAML SSO Configuration"

33.4.2.1 SAML Single Sign-on Prerequisites

This section describes a set of steps that should be carried out prior to configuring SAML-based single sign-on. Note that these steps assume that Spaces and associated components are already installed and the relevant connections have been configured and tested.

The prerequisites for SAML-based SSO are described in the following subsections:

- Section 33.4.2.1.1, "Configuring Oracle Content Server for SAML SSO"
- Section 33.4.2.1.2, "Configuring the Discussions Server for SAML SSO"
- Section 33.4.2.1.3, "Configuring and Exporting the Certificates"
- Section 33.4.2.1.4, "Setting Up SSL"

33.4.2.1.1 Configuring Oracle Content Server for SAML SSO

If your instance uses a Documents connection that requires the use of OHS to surface the Content Server user interface in WebCenter Portal, you need to configure WebCenter Portal and related applications with a WebTier.

When configuring SAML SSO for a configuration that includes Content Server, all HTTP URLs should point to the WebTier host and port. Additionally, when Content Server is front-ended with OHS, the following entries must appear in `mod_wl_ohs.conf`, apart from the usual configuration for WebCenter Portal:

```
<Location /cs>
      SetHandler weblogic-handler
      WebLogicHost ucm.example.com
      WebLogicPort 16200
</Location>

<Location /adfAuthentication>
      SetHandler weblogic-handler
      WebLogicHost ucm.example.com
      WebLogicPort 16200
</Location>

<Location /samlacs/acs>
      SetHandler weblogic-handler
```

```
      WebLogicHost ucm.example.com
      WebLogicPort 16200
</Location>
```

See Section 33.2.5, "Installing and Configuring the Oracle HTTP Server" for more information about installing OHS and editing `mod_wl_ohs.conf`.

Additionally, when a custom login page is used for WebCenter Portal the following HTML comment must be added to the head section of the HTML page generated for Content Server for Site Studio Designer to work:

```
<!--IdcClientLoginForm=1-->
```

This HTML comment appears in the out-of-the-box log in pages in WebCenter Portal, but if you configure a new page to be the login page in a SAML SSO setup, then the comment must be added by hand, or in generated HTML as shown in the following example for a JSF page:

```
<af:document id="d1">
  <f:facet name="metaContainer">
      <f:verbatim>
        ${cb.commentText}
      </f:verbatim>
  </f:facet>
  .........
```

where `cb` is a managed bean containing the method:

```
public String getCommentText(){
   return "<!--IdcClientLoginForm=1-->";
 }
```

After checking that the comment text is added verbatim in the `metaContainer` facet of `af:document`, check the generated HTML page using View Source and confirm that `<!--IdcClientLoginForm=1-->` is in the `<head>` section of the HTML page.

33.4.2.1.2 Configuring the Discussions Server for SAML SSO

By default, the .EAR file that is deployed for the Oracle WebCenter Portal's Discussion Server Server supports form-based Oracle SSO or Oracle Access Manager SSO. Therefore, before you can configure the Oracle WebCenter Portal's Discussion Server Server for SAML-based single sign-on, you must also first deploy the SAML SSO version of the discussion server .EAR file.

> **Note:** Before configuring the discussions server for SSO, ensure that it is configured to use the same identity store LDAP as WebCenter Portal, as described in Section 31.1, "Reassociating the Identity Store with an External LDAP Server." If you've chosen not to move the default administrator account to an external LDAP, be sure to also follow the instructions in Section 31.4.1, "Migrating the Discussions Server to Use an External LDAP."

To deploy and configure the SAML SSO version of the Oracle WebCenter Portal's Discussion Server:

1. Log in to the WebLogic Server Administration Console as an administrator.

   For information on logging in to the WebLogic Server Administration Console, see Section 1.13.2, "Oracle WebLogic Server Administration Console."

2. In the Domain Structure pane, click **Deployments**.

The Deployments Summary pane displays.

3. On the Deployment Summary page, select `owc_discussions stop and delete` and click **Install**.
4. Using the Install Application Assistant **Path** field, locate the SSO enabled owc_discussions .EAR file (`owc_discussions_samlsso.ear`, typically in *`WCP_ORACLE_HOME`*`/discussionserver`).
5. Select the `owc_discussions_samlsso.ear` file and click **Next**.
6. Select **Install this deployment as an application** and click **Next**.
7. Set the **Name** to `owc_discussions`.
8. Deploy the .EAR file.
9. Log in to the Discussions Server Administration Console as an administrator (see Section 33.2.6.2, "Configuring the Discussions Server for SSO" for more information on logging in to the Discussions Server Administration Console).
10. Open the System Properties page and edit (if it already exists) or add the `owc_discussions.sso.mode` property, setting its value to `true`.
11. Restart the `WC_Collaboration` managed server (where the discussions server is deployed).

33.4.2.1.3 Configuring and Exporting the Certificates

To secure communication between the SAML source and destination sites, communication should be encrypted. Additionally, certificates should be used to verify the identity of the other party during SAML interaction. Follow the steps below to generate a key using the `keytool` utility (available as part of the JDK 6.0), and register it using the WebLogic Server Administration Console.

To create certificates using keytool:

1. Configure the necessary keystore for the `WC_Spaces` and Administration servers in the WebCenter Portal domain. This keystore should contain the certificate you intend to use for securing the SAML assertions.

   If you only want to test the configuration, you can either create a "demoidentity" certificate that is packaged in the `DemoIdentity` keystore that is configured by default, or you can use `keytool` to generate a new certificate in the `DemoIdentity` keystore. For more information about configuring a custom identity keystore, see Section 35.1.2, "Configuring the Custom Identity and Java Trust Keystores."

2. Using `keytool`, export the certificate you have chosen to use to encrypt SAML assertions. Be sure to run the `export` command on the keystore that is configured for `WC_Spaces` and the Administration server for the WebCenter Portal domain.

   ```
   keytool -export -keystore <keystore_name> -storepass <keystore_password> -alias
   <certificate_alias> -keypass <certificate_password> -file <certificate.der>
   ```

   where:

   - *`keystore_name`* is the name of the keystore that is configured for `WC_Spaces` and the Administration server for the WebCenter Portal domain
   - *`keystore_password`* is the password of the keystore that is configured for `WC_Spaces` and the Administration server for the WebCenter Portal domain
   - *`certificate_alias`* is the alias name (for example, `demoidentity`)

- *certificate_password* is the password for the certificate
- *certificate.der* is the name of the certificate file

Note that the `keytool -export` command should be run from the WebCenter Portal machine and should export the certificate being used in the SAML SSO setup residing in the keystore configured for the WebCenter Portal server.

3. Copy or transfer the file into the destination domains (such as SOA, Content Server, and Collaboration) and configure the `certPath` value in the `wcsamlsso.properties` file when you are ready to run the SAML SSO script as described in Section 33.4.2.2.1, "The Single Sign-on Script."

33.4.2.1.4 Setting Up SSL

If the WebCenter Portal installation requires SSL for providing transport-level security, then SSL should be configured before configuring single sign-on as described in Chapter 35, "Configuring SSL." Note that setting up SSL is not related to enabling SSO.

33.4.2.2 Configuring SAML-based SSO

After installing WebCenter Portal and services and components as required for your environment, continue by configuring SAML-based single sign-on using the scripts as described in this section.

The scripts set up SAML-based single sign-on in a WebLogic environment by configuring:

- SAML Credential Mapping Provider
- Necessary relying parties
- Source Site Federation Services
- SAML Identity Asserter
- Necessary asserting parties
- Destination Site Federation Services

This section includes the following subsections:

- Section 33.4.2.2.1, "The Single Sign-on Script"
- Section 33.4.2.2.2, "Using the Scripts"

33.4.2.2.1 The Single Sign-on Script

The single sign-on script to configure SAML 1.1 SSO for WebCenter Portal and related applications is located in the *WCP_ORACLE_HOME* `/webcenter/scripts/samlsso` folder. The following files are relevant for SAML configuration:

- wcsamlsso.properties
- wcsamlsso.py
- configureSpaces.py
- configureCollab.py
- configureUtilities.py
- configureSOA.py
- configureUCM.py
- configureREST.py

- configureForum.py
- configureActivityGraphEngine.py
- configureWorklistIntegration.py
- configureWorklistDetail.py
- configureWorklistSDP.py
- configureCS.py

wcsamlsso.properties

This properties file (*WCP_ORACLE_HOME*/common/bin/wcsamlsso.properties) encapsulates the necessary configuration information for the SAML SSO setup. The properties file has the following sections:

spaces_config

This section captures the login information, WebLogic Admin URL, WebCenter Portal server and URL, and so forth, of the WebCenter Portal domain required for the Credential Mapper and Source Site Federation Services configuration. All properties in this section must be completed.

- configFile - Config file containing the weblogic user account and password for the WebCenter Portal domain
- keyFile - Key file to decrypt the weblogic user account and password for the WebCenter Portal domain
- adminURL - WebLogic Admin URL to connect to WLST
- usesSSL - Indicates whether WebCenter Portal is configured to use SSL
- url - WebCenter Portal URL. If usesSSL is "true", then change "http" to "https". If WebCenter Portal is front-ended with WebTier, then specify the WebTier host and port here.
- serverName - Server where WebCenter Portal is deployed, typically WC_Collaboration
- certAlias - Alias of certificate to sign SAML assertions
- certPassword - Encrypted password of certificate to sign SAML assertions

collab_config

This section captures the login information, admin URL, certificate file path, and so forth, of the Collaboration domain required for the Identity Asserter and Destination Site Federation Services configuration. Only complete this section if your setup has discussions configured.

- configFile - Config file containing weblogic user account and password for the Services domain
- keyFile - Key file to decrypt weblogic user account and password for the Services domain
- adminURL - WebLogic Admin URL to connect to WLST
- usesSSL - Indicates whether discussions is configured to use SSL
- serverName - Server where discussions is deployed (typically the WC_Collaboration managed server)
- certAlias - Alias of certificate to verify SAML assertions

- `certPath` - Path to exported certificate to verify SAML assertions. Note that the certificate path should be a valid path on the machine that hosts the domain (i.e., the one specified in `adminURL`)

`utilities_config`

This section captures the login information, admin URL, and certificate file path of the Utilities domain required for the Identity Asserter and Destination Site Federation Services configuration. Complete this section out only if your setup is configured with the Activity Graph application.

- `configFile` - Config file containing `weblogic` user account and password for the Utilities domain
- `keyFile` - Key file to decrypt `weblogic` user account and password for the Utilities domain
- `adminURL` - WebLogic Admin URL to connect to WLST
- `usesSSL` - Indicates whether Utilities applications are configured to use SSL
- `serverName` - Server where Utilities applications are deployed (typically the `WC_Utilities` managed server)
- `certAlias` - Alias of certificate to verify SAML assertions
- `certPath` - Path to exported certificate to verify SAML assertions. Note that the certificate path should be a valid path on the machine that hosts the domain (i.e., the one specified in `adminURL`)

`soa_config`

This section captures the login information, admin URL, certificate file path, and so forth, of the SOA domain required for the Identity Asserter and Destination Site Federation Services configuration. Only complete this section if your setup has SOA configured.

- `configFile` - Config File containing the `weblogic` user account and password for the SOA domain
- `keyFile` - Key File to decrypt the `weblogic` user account and password for the SOA domain
- `adminURL` - WebLogic Admin URL to connect to WLST
- `usesSSL` - Indicates whether SOA applications are configured to use SSL
- `serverName` - Server where SOA applications are deployed (typically `soa_server1`)
- `certAlias` - Alias of certificate to verify SAML assertions
- `certPath` - Path to exported certificate to verify SAML assertions. Note that the certificate path should be a valid path on the machine that hosts the domain (i.e., the one specified in `adminURL`)

`ucm_config`

This section captures the login information, admin URL, certificate file path, and so forth, of the Content Server domain required for the Identity Asserter and Destination Site Federation Services configuration. Only complete this section if your installation has the Documents service configured.

- `configFile` - Config file containing the weblogic user name and password for the Content Server (UCM) domain

- usesSSL - Indicates whether Content Server applications are configured to use SSL
- keyFile - Key File to decrypt the weblogic user account and password for the Content Server (UCM) domain
- adminURL - WebLogic Administration URL to connect to WLST
- serverName - Server where Content Server applications are deployed (typically UCM_server1)
- certPath - Path to exported certificate to verify SAML assertions. Note that the certificate path should be a valid path on the machine that hosts the domain (i.e., the one specified in adminURL)

rss_config

This is mandatory

- url - RSS URL. If usesSSL in **spaces_config** is "true", then change "http" to "https". If RSS is front-ended with WebTier, then specify the WebTier host and port here.

rest_config

This section must be completed.

- url - REST URL. If usesSSL in **spaces_config** is "true", then change "http" to "https". If REST is front-ended with WebTier, then specify the WebTier host and port here.

forum_config

Complete this section if your configuration has discussions installed.

- url - OWC discussions URL. If usesSSL in **collab_config** is "true", then change "http" to "https". If discussions is front-ended with WebTier, then specify the WebTier host and port here.

worklist_config

Complete this section of SOA is installed and Worklist is configured for WebCenter Portal.

- worklist_detail - Worklist Detail application URL. If usesSSL in **soa_config** is "true", then change "http" to "https". If Worklist Detail application is front-ended with WebTier, then specify the WebTier host and port here.
- worklist_sdp - Worklist SDP application URL. If usesSSL in **soa_config** is "true", then change "http" to "https". If Worklist Detail application is front-ended with WebTier, then specify the WebTier host and port here.
- worklist_integration - Worklist Integration application URL. If usesSSL in **soa_config** is "true", then change "http" to "https". If Worklist Detail application is front-ended with WebTier, then specify the WebTier host and port here.

activitygraph_config

Complete this section if your configuration has the Utilities server installed.

- url - ActivityGraphEngines URL. If usesSSL in **spaces_config** is "true", then change "http" to "https". If the Activity Graph application is front-ended with WebTier, then specify the WebTier host and port here.

cs_config

Complete this section if your configuration has Content Server installed and you have a documents connection configured for the WebCenter Portal application.

- `url` - Content Server URL. If `usesSSL` in **spaces_config** is "`true`", then change "`http`" to "`https`". If Content Server is front-ended with WebTier, then specify the WebTier host and port here. Note that if both WebCenter Portal and Content Server are configured for your environment, then they must both be accessed using the same WebTier.

wcsamlsso.py

Script file (*WCP_ORACLE_HOME*`/common/wlst/wcsamlsso.py`) that contains utility functions invoked by rest of the configuration scripts

configureSpaces.py

Executable script (*WCP_ORACLE_HOME*`/webcenter/scripts/samlsso/configureSpaces.py`) to configure SAML 1.1 Credential Mapper, SAML 1.1 Identity Asserter and Source and Destination site federation services on the WebCenter Portal domain

configureCollab.py

Executable script (*WCP_ORACLE_HOME*`/webcenter/scripts/samlsso/configureCollab.py`) to configure SAML 1.1 Identity Asserter and Destination site federation services on the Collaboration domain

configureUtilities.py

Executable script (*WCP_ORACLE_HOME*`/webcenter/scripts/samlsso/configureUtilities.py`) to configure SAML 1.1 Identity Asserter and Destination site federation services on the Utilities domain

configureSOA.py

Executable script (*WCP_ORACLE_HOME*`/webcenter/scripts/samlsso/configureSOA.py`) to configure SAML 1.1 Identity Asserter and Destination site federation services on the SOA domain

configureUCM.py

Executable script (*WCP_ORACLE_HOME*`/webcenter/scripts/samlsso/configureUCM.py`) to configure SAML 1.1 Identity Asserter and Destination site federation services on the Content Server domain

configureREST.py

Executable script (*WCP_ORACLE_HOME*`/webcenter/scripts/samlsso/configureREST.py`) to configure asserting and relying parties for the REST application

configureRSS.py

Executable script (*WCP_ORACLE_HOME*`/webcenter/scripts/samlsso/configureRSS.py`) to configure asserting and relying parties for RSS

application

configureForum.py

Executable script (*WCP_ORACLE_HOME*/webcenter/scripts/samlsso/configureForum.py) to configure asserting and relying parties for discussions

configureActivityGraphEngine.py

Executable script (*WCP_ORACLE_HOME*/webcenter/scripts/samlsso/configureActivityGraphEngine.py) to configure asserting and relying parties for the Activity Graph Engine application

configureWorklistIntegration.py

Executable script (*WCP_ORACLE_HOME*/webcenter/scripts/samlsso/configureWorklistIntegration.py) to configure asserting and relying parties for the Worklist Integration application

configureWorklistDetail.py

Executable script (*WCP_ORACLE_HOME*/webcenter/scripts/samlsso/configureWorklistDetail.py) to configure asserting and relying parties for the Worklist Community Detail application

configureWorklistSDP.py

Executable script (*WCP_ORACLE_HOME*/webcenter/scripts/samlsso/configureWorklistSDP.py) to configure asserting and relying parties for the Worklist SDP application

configureCS.py

Executable script (*WCP_ORACLE_HOME*/webcenter/scripts/samlsso/configureCS.py) to configure asserting and relying parties for the Content Server application.

33.4.2.2.2 Using the Scripts

Follow the steps below to use the scripts to configure SAML-based single sign-on:

> **Note:** If you encounter errors when running the scripts due to configuration errors, the SAML SSO configuration may be left in an incomplete state. The config scripts provided are not re-runnable; you must clean up the SAML SSO artifacts before you retry the configuration as described in Section 33.4.2.6, "Removing Your SAML SSO Configuration."

1. Ensure that the Administration server for all the domains used in this configuration are up and running.
2. Generate the config and key files containing the connection information for the various domains using the `storeUserConfig` WLST command from the *WCP_ORACLE_HOME*/common/bin so that the properties file is picked up. Use the command-line help (`help('storeUserConfig')`) for usage and syntax details.
   a. Connect using WLST to the WebCenter Portal domain using the admin username and password, and run the following command:

```
storeUserConfig('spacesconfig.secure', 'spaceskey.secure')
```

This creates a user configuration file and an associated key file. The user configuration file contains an encrypted username and password. The key file contains a secret key that is used to encrypt and decrypt the username and password. The above command stores the config and key files in the directory from where WLST was invoked, or you can optionally specify a more secure path.

b. Repeat step **2a** after connecting to the Collaboration domain using the admin username and password. Even if the Utilities server is in the same domain as WebCenter Portal (`wc_domain`), you must connect to the WebCenter Portal domain and run this command:

```
storeUserConfig('collabconfig.secure',
'collabkeykey.secure')
```

c. Repeat step **2a** after connecting to the Utilities domain using the admin username and password. Even if the Utilities server is in the same domain as WebCenter Portal (`wc_domain`), you must connect to the WebCenter Portal domain and run this command:

```
storeUserConfig('utilitiesconfig.secure',
'utilitieskey.secure')
```

d. Repeat step **2a** after connecting to the SOA domain using the admin username and password. Even if SOA is installed on the same domain as WebCenter Portal, you must connect to the WebCenter Portal domain and run this command:

```
storeUserConfig('soaconfig.secure', 'soakey.secure')
```

e. Repeat step **2a** after connecting to the Content Server domain using the admin username and password.

```
storeUserConfig('ucmconfig.secure', 'ucmkey.secure')
```

3. Launch WLST and run the WLST `encrypt` command to encrypt the certificate password. Use the command-line help (`help('encrypt')`) for usage and syntax details.

```
print encrypt(obj='<certificatePassword>', domainDir='<full
path to the WebCenter Portal domain directory>')
```

This displays the encrypted certificate password. The encrypt command uses the encryption for a specified WebLogic Server domain root directory. The encrypted output needs to be set as the `certPassword` value in `wcsamlsso.properties` mentioned in the next step. Since this password will be set onto the credential mapper and source site federation services in the WebCenter Portal domain, ensure that you run the encryption utility from the WebCenter Portal domain.

4. Edit *`WCP_ORACLE_HOME`*`/common/bin/wcsamlsso.properties` and complete the sections applicable to your setup. Refer to Section 33.4.2.2.1, "The Single Sign-on Script" for a detailed description of the sections in the properties file.

5. Launch WLST from *`WCP_ORACLE_HOME`*`/common/bin` and execute the scripts in the order shown below.

Note: Run the scripts in the WLST offline mode as the scripts include an explicit connect command.

---

a. `execfile('<WCP_ORACLE_HOME>/webcenter/scripts/samlsso/configureSpaces.py')`

Restart all servers including the Administration server in the WebCenter Portal domain.

b. If you have a discussions server set up, execute the `configureCollab.py` script:

```
execfile('<WCP_ORACLE_HOME>/webcenter/scripts/samlsso/configureCollab.py')
```

If discussions belongs to the same domain as WebCenter Portal, then only restart the `WC_Collaboration` managed server. Otherwise, restart all servers including the Administration server in the Collaboration domain.

c. If you have a Utilities server set up, execute the `configureUtilities.py` script:

```
execfile('<WCP_ORACLE_HOME>/webcenter/scripts/samlsso/configureUtilities.py')
```

If the Utilities server belongs to the same domain as WebCenter Portal, then only restart the `WC_Utilities` server. Otherwise, restart all servers including the Administration server in the Utilities domain.

d. If you have Worklist configured for WebCenter Portal, execute the `configureSOA.py` script:

```
execfile('<WCP_ORACLE_HOME>/webcenter/scripts/samlsso/configureSOA.py')
```

Restart all servers including the Administration server in the SOA domain.

e. If you have documents configured for WebCenter Portal, run the `configureUCM.py` script as shown below:

```
execfile('WCP_ORACLE_HOME/webcenter/scripts/samlsso/configureUCM.py')
```

Restart all servers including the Administration server in the Content Server domain.

6. Run the individual commands below as required for your environment.

`execfile('<WCP_ORACLE_HOME>/webcenter/scripts/samlsso/configureREST.py')` - No restart is required.

`execfile('<WCP_ORACLE_HOME>/webcenter/scripts/samlsso/configureRSS.py')` - No restart is required.

`execfile('<WCP_ORACLE_HOME>/webcenter/scripts/samlsso/configureForum.py')` - Do this if you have discussions installed in your setup. No restart is required.

`execfile('<WCP_ORACLE_HOME>/webcenter/scripts/samlsso/configureActivityGraphEngine.py')` - Do this if you have Utilities installed in your setup. No restart is required.

`execfile('<WCP_ORACLE_HOME>/webcenter/scripts/samlsso/configureWorklistIntegration.py')` - Do this if you have Worklist installed in your setup. No restart is required.

`execfile('<WCP_ORACLE_HOME>/webcenter/scripts/samlsso/configureWorklistDetail.py')` - Do this if you have Worklist installed in your setup. No restart is required.

`execfile('<WCP_ORACLE_HOME>/webcenter/scripts/samlsso/configureWorklistSDP.py')` - Do this if you have Worklist installed in your setup. No restart is required.

`execfile('<WCP_ORACLE_HOME>/webcenter/scripts/samlsso/configureCS.py')` - Do this if you have Content Server installed in your setup. No restart is required.

7. Check your installation using the steps provided in Section 33.4.2.4, "Checking Your Configuration."

IMPORTANT: Since the properties file contains sensitive information, delete it from `<WCP_ORACLE_HOME>/common/bin` after you have configured and verified the SAML SSO setup. Also delete the config and key files you generated in **step 2** above.

Note: If you encounter errors when running the scripts, you must remove the asserting and relying parties set up by the scripts before running the scripts again as described in Section 33.4.2.6, "Removing Your SAML SSO Configuration."

After removing your old SAML SSO configuration, continue by re-running the scripts.

33.4.2.3 Configuring SAML SSO for RSS Using External Readers

By default, WebCenter Portal RSS feeds are protected by SSO. However, they will not work well with external readers if left protected. If access using external readers is important, Oracle recommends that the WebCenter Portal RSS resource be unprotected so that the authentication for the RSS Servlet is handled by WebLogic Server's BASIC authentication that external readers can handle.

Follow the steps below to unprotect the RSS feeds:

1. Log onto the WLS Administration Console for the WebCenter Portal domain.
2. Open the security realm and select **Providers >Credential Mapping > wcsamlcm> Management > Relying Parties**.
3. Disable or delete the relying party for RSS.
4. Open the security realm and select **Providers > Authentication > wcsamlia > Management > Asserting Parties**.
5. Disable or delete the asserting party for RSS.

33.4.2.4 Checking Your Configuration

Follow the steps below to check that your single sign-on configuration is working correctly.

To test your single sign-on configuration:

1. Using a new browser, log in to WebCenter Portal and check that you're not challenged for credentials when you click **Forum Administration** from **Space Settings > Services > Discussions** (assuming this service is provisioned for the space).
2. Access the RSS link from the discussions or workList task flow and check that you are not challenged to log in.

3. For Content Server, go to the Profile user interface and make sure you see Content Server screens embedded in iFrames without being challenged to log in. You should also be able to access Site Studio content in Content Presenter templates without being challenged to log in as you are already logged into WebCenter Portal.
4. Access `http://host:port/rest/api/resourceIndex` and make sure you see the BASIC authentication challenge. If you are already logged in to another related application that uses the same SSO, you should shown content without being challenged to log in.
5. To test SOA, access links in the Workflow task flow and make sure you are not challenged to log in.

If while testing SAML SSO you encounter 404 or 403 errors, check the SAML configuration and also turn on debug logging for SAML on the `AdminServer`. Also turn on logging for the `WC_Spaces` server and the server hosting your destination site. The logs will be available in `$domain.home/servers/<server>/logs/<server>.log`. For information on how to turn on logging for `WC_Spaces` and other application servers, see Section 28.2.1, "Viewing and Configuring WebCenter Portal Logs." Before re-running the scripts, remove your SAML SSO configuration as described in Section 33.4.2.6, "Removing Your SAML SSO Configuration."

33.4.2.5 Disabling Your SAML SSO Configuration

This section describes how to temporarily disable your SAML SSO configuration for testing or other purposes.

To disable your SAML SSO configuration:

1. Log onto the WLS Administration Console for the WebCenter Portal domain.
2. Open the security realm and select **Providers >Credential Mapping > wcsamlcm> Management > Relying Parties** and disable all the relying parties shown there.
3. Open the security realm and select **Providers > Authentication > wcsamlia > Management > Asserting Parties** and disable all the asserting parties shown there.
4. If there are other WLS domains, such as SOA or Content Server, that have been configured with SAML SSO, remove the SAML SSO configuration from these domains as well:
   a. Log in to the WLS Administration Console for the WLS domain.
   b. Open the security realm and select **Providers > Authentication > wcsamlia > Management > Asserting Parties** and disable all the asserting parties shown there.
5. Confirm that the SAML SSO configuration has been disable by opening your applications and checking that you are not prompted to sign in.

33.4.2.6 Removing Your SAML SSO Configuration

Since the SAML SSO configuration scripts do not include a cleanup facility, if you have made errors while updating the `wcsamlsso.properties` file or running the scripts, the configuration could be in an invalid state. At this point, it's better to clean up all the SAML SSO configurations and start over. This section describes the steps to remove the SAML SSO configuration.

Note that if you have fully set up SAML SSO (i.e., the script ran to completion), then all the instructions below will be valid. However, if you encountered errors while

running the script, then the configuration may be incomplete and only some of the artifacts below will be present and will need to be removed.

To remove your SAML SSO configuration:

1. Log onto the WLS Administration Console for the WebCenter Portal domain.
2. Open the security realm and select **Providers >Credential Mapping > wcsamlcm> Management > Relying Parties** and delete all the relying parties shown there.
3. Open the security realm and select **Providers > Authentication > wcsamlia > Management > Asserting Parties** and delete all the asserting parties shown there.
4. Go to **Providers > Authentication > wcsamlia > Management > Certificates** and delete the certificate there.
5. Go to **Providers > Credential Mapping > wcsamlcm** and delete the SAML Credential Mapper.
6. Go to **Providers > Authentication > wcsamlia** and delete the SAML Identity Asserter.
7. Restart the entire WebCenter Portal WLS domain.
8. If there are other WLS domains, such as SOA or Content Server, that have been configured with SAML SSO, remove the SAML SSO configuration from these domains as well:
   a. Log in to the WLS Administration Console for the WLS domain.
   b. Open the security realm and select **Providers > Authentication > wcsamlia > Management > Asserting Parties** and delete all the asserting parties shown there.
   c. Go to **Providers > Authentication > wcsamlia > Management > Certificates** and delete the certificate there.
   d. Go to **Providers > Authentication > wcsamli**a and delete the SAML Identity Asserter.
   e. Restart the entire WLS domain.
9. Confirm that the SAML SSO configuration has been removed by opening your applications and checking that you are not prompted to sign in. You can now safely use the scripts again to reconfigure SAML SSO.

33.5 Configuring SSO for Microsoft Clients

This section describes how to set up single sign-on (SSO) for Microsoft clients, using Windows authentication based on the Simple and Protected Negotiate (SPNEGO) mechanism and the Kerberos protocol, together with the WebLogic Negotiate Identity Assertion provider for WebCenter Portal. This SSO approach enables Microsoft clients (such as browsers), authenticated in a Windows domain using Kerberos, to be transparently authenticated to web applications (such as WebCenter Portal) in a WebLogic domain based on the same credentials, and without the need to type in their password again. For more information about using Microsoft Office clients with WebCenter Portal, see Chapter 26, "Managing Microsoft Office Integration."

Cross-platform authentication is achieved by emulating the negotiate behavior of native Windows-to-Windows authentication services that use the Kerberos protocol. In order for cross-platform authentication to work, non-Windows servers (in this case, WebLogic Server) must parse SPNEGO tokens in order to extract Kerberos tokens, which are then used for authentication.

This section contains the following subsections:

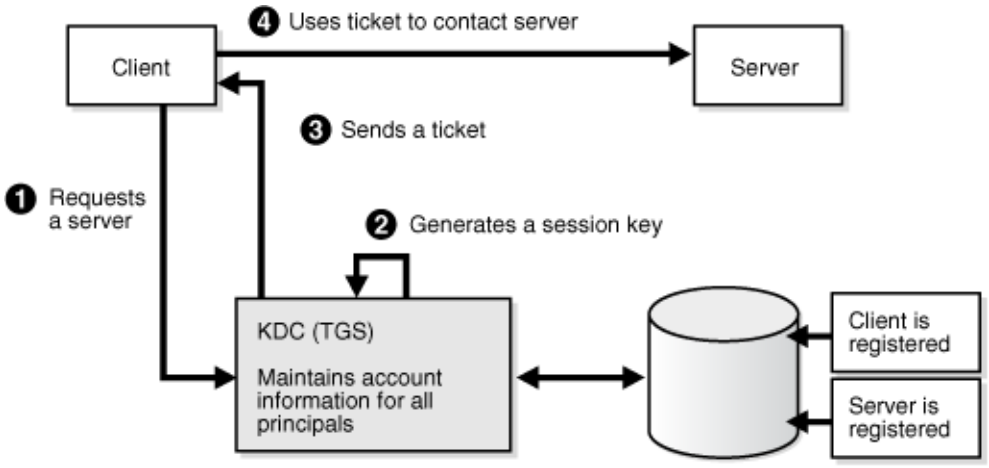
33.5.1 Microsoft Client SSO Concepts

Understanding Kerberos

Kerberos is a secure method for authenticating a request for a service in a network. The Kerberos protocol comprises three parties: a client, a server and a trusted third party to mediate between them, known as the KDC (Key Distribution Center). Under Kerberos, a server allows a user to access its service if the user can provide the server a Kerberos ticket that proves its identity. Both the user and the service are required to have keys registered with the KDC.

The diagram below describes the basic exchanges that must take place before a client connects to a server.

Figure 33–10 Connecting to a Server Through a Key Distribution Center



Understanding SPNEGO

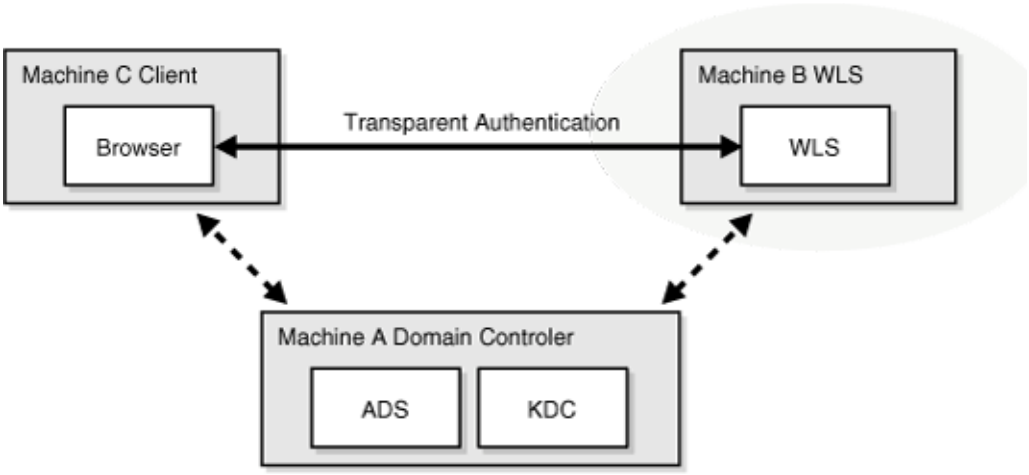
SPNEGO (Simple and Protected GSSAPI Negotiation Mechanism) is a GSSAPI "pseudo mechanism" that is used to negotiate one of several possible real mechanisms. SPNEGO is used when a client application wants to authenticate to a remote server, but neither end is sure what authentication protocols the other supports. The pseudo-mechanism uses a protocol to determine what common GSSAPI mechanisms are available, selects one, and then dispatches all further security operations to it. This can help organizations deploy new security mechanisms in a phased manner.

SPNEGO's most visible use is in Microsoft's HTTP Negotiate authentication extension. The negotiable sub-mechanisms include NTLM and Kerberos, both used in Active Directory.

This feature enables a client browser to access a protected resource on WebLogic Server, and to transparently provide the WebLogic Server with authentication information from the Kerberos database using a SPNEGO ticket. The WebLogic Server can recognize the ticket and extract the information from it. WebLogic Server then uses the information for authentication and grants access to the resource if the

authenticated user is authorized to access it. (Kerberos is responsible for authentication only; authorization is still handled by WebLogic Server.)

Figure 33–11 SPNEGO-based Authentication



33.5.2 System Requirements

To use SSO with Microsoft clients you need:

A host computer with:

- Windows 2000 or later installed
- Fully-configured Active Directory authentication service. Specific Active Directory requirements include:
  - User accounts for mapping Kerberos services
  - Service Principal Names (SPNs) for those accounts
  - Key tab files created and copied to the start-up directory in the WebLogic Server domain
- WebLogic Server installed and configured properly to authenticate through Kerberos, as described in this section

Client systems with:

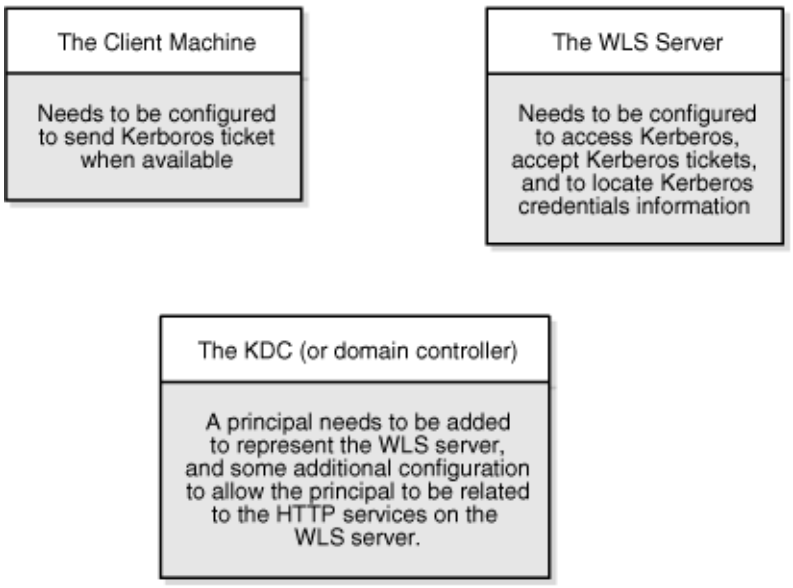
- Windows 2000 Professional SP2 or later installed
- One of the following types of clients:
  - A properly configured Internet Explorer browser. Internet Explorer 6.01 or later is supported.
  - .NET Framework 1.1 and a properly configured Web service client.

Note: Clients must be logged on to a Windows 2000 domain and have Kerberos credentials acquired from the Active Directory server in the domain. Local logons will not work.

33.5.3 Configuring Microsoft Clients

Configuring SSO with Microsoft clients requires configuring the Microsoft Active Directory, the Microsoft client, and the WebLogic Server domain shown in Figure 33–12. For detailed configuration steps and troubleshooting, see the "Configuring Single Sign-On with Microsoft Clients" section in *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

Figure 33–12 Configuring SSO with Microsoft Clients



To configure Microsoft clients for SSO:

1. Configure your network domain to use Kerberos.
2. Create a Kerberos identification for WebLogic Server.
   a. Create a user account in the Active Directory for the host on which WebLogic Server is running.
   b. Create a Service Principal Name for this account.
   c. Create a user mapping and keytab file for this account (see the "Configuring Single Sign-On with Microsoft Clients" section in *Oracle Fusion Middleware Securing Oracle WebLogic Server*).
3. Choose a browser client (Internet Explorer or Mozilla Firefox) and configure it to use Kerberos tokens (see the "Enabling the Browser to Return Kerberos Tokens" section in *Oracle Fusion Middleware Integration Guide for Oracle Access Manager*).
4. Set up the WebLogic Server domain (`wc_domain` in this case) to use Kerberos authentication.
   a. Create a JAAS login file that points to the Active Directory server in the Microsoft domain and the keytab file created in Step 2 (see the "Creating a JAAS Login File" section in *Oracle Fusion Middleware Securing Oracle WebLogic Server*).
   b. Configure a Negotiate Identity Assertion provider in the WebLogic Server security realm (see Section 33.5.3.1, "Configuring the Negotiate Identity Assertion Provider").

c. Configure the WebLogic Server domain to use the Active Directory Authenticator so that the WebLogic domain uses the same Active Directory of the domain as the identity store. You could also use a different identity store and match the users in this store with the Active Directory users of your domain, but using the Active Directory authenticator is recommended as maintaining two different identity stores risks them getting out of sync (see Section 33.5.3.2, "Configuring an Active Directory Authentication Provider").

Caution: Ensure that only the identity store is configured for Active Directory. The policy and credential stores are not certified for Active Directory.

---

5. Add the following system properties to the `JAVA_OPTIONS` in `setDomainEnv.sh` for each WebCenter Portal machine, changing the values below for the values of the particular host (on one line):

```
-Dnon_sso_protocol=http (the protocol to access WebCenter Portal directly
through the WC_Spaces server without going through OHS)
-Dnon_sso_host=example.com (the host for the WLS WC_Spaces server)
-Dnon_sso_port=8888 (the port for the WLS WC_Spaces server)
-Dsso_base_url=http://example.com:7777 (the URL for accessing the WC_Spaces
server through OHS)
```

   The `non_sso` values are the value on the machine for protocol, host, and port. The `sso` values are the value that the user would see when directed through OHS.

6. For WebCenter Portal, configure the WebTier OHS so that it forwards requests to the Oracle WebLogic Server for WebCenter Portal, as described in Section 33.6, "Configuring SSO with Virtual Hosts."

7. Restart the WebLogic Servers (Administration Server and managed servers) using the startup arguments specified in step 5. Repeat steps 4, 5, and 6 for the SOA domain to enable single sign-on for SOA applications.

8. Restart the OHS for the changes to take effect.

9. Configure the discussions server (see Section 33.5.3.4, "Configuring the Discussions Server for SSO").

33.5.3.1 Configuring the Negotiate Identity Assertion Provider

This section provides instructions for creating and configuring a Negotiate Identity Assertion provider. The Negotiate Identity Assertion provider enables single sign-on (SSO) with Microsoft clients. The identity assertion provider decodes Simple and Protected Negotiate (SPNEGO) tokens to obtain Kerberos tokens, validates the Kerberos tokens, and maps them to WebLogic users. The Negotiate Identity Assertion provider uses the Java Generic Security Service (GSS) Application Programming Interface (API) to accept the GSS security context through Kerberos.

To configure the Negotiate Identity Assertion provider:

1. Log in to the WebLogic Server Administration Console.

   For information on logging in to the WebLogic Server Administration Console, see Section 1.13.2, "Oracle WebLogic Server Administration Console."

2. From the Domain Structure pane, click **Security Realms**.

   The Summary of Security Realms pane displays.

3. Click your security realm.

   The Settings page for the security realm displays.

4. Open the **Providers** tab and select the **Authentication** subtab.

   The Authentication Settings pane displays.

5. Click **New**.

   The Create a New Authentication Provider pane displays.

6. Enter a **Name** for the identity asserter, and select `NegotiateIdentityAsserter` as the **Type**.

7. Click **OK**.

33.5.3.2 Configuring an Active Directory Authentication Provider

Follow the steps below to configure an Active Directory authentication provider using the WebLogic Administration Console.

To configure an Active Directory Authentication provider:

1. Log in to the WebLogic Server Administration Console.

   For information on logging in to the WebLogic Server Administration Console, see Section 1.13.2, "Oracle WebLogic Server Administration Console."

2. From the Domain Structure pane, click **Security Realms**.

   The Summary of Security Realms pane displays.

3. Click your security realm.

   The Settings page for the security realm displays.

4. Open the **Providers** tab and select the **Authentication** subtab.

   The Authentication Settings pane displays.

5. Click **New**.

   The Create a New Authentication Provider pane displays.

6. Enter a **Name** for the authentication provider, and select `ActiveDirectoryAuthenticator` as the **Type**.

7. Click **OK**.

8. Click the authentication provider you just created in the list of providers.

   The Settings page for the provider displays.

9. Open the **Configuration** tab and the **Common** subtab.

10. Set the Control Flag to `SUFFICIENT` and click **Save**.

> **Note:** The Control Flag settings of any other authenticators must also be changed to `SUFFICIENT`. If there is a pre-existing Default Authenticator that has its Control Flag set to `REQUIRED`, it must be changed to `SUFFICIENT`.

11. Open the **Provider Specific** subtab.

    The Provider Specific Settings pane displays.

12. Complete the fields as shown in the table below. Leave the rest of the fields set to their default values.

Table 33–3 Active Directory Authenticator Settings

Parameter	Value	Description
Host:		The host ID of the LDAP server
Port:		The port number of the LDAP server
Principal:		The LDAP administrator principal
Credential:		
User Base DN:		The user search base (for example, OU=spnego unit,DC=admin,DC=oracle,DC=com)
User From Name Filter:	(&(cn=%u)(objectclass=user))	
User Search Scope:	subtree	
User Name Attribute:	cn	
User Search Scope:	user	
Group Base DN:		The group search base (same as User Base DN)
Group From Name Filter:	(&(cn=%g)(objectclass=group))	
Group Search Scope:	subtree	
Static Group Name Attribute:	cn	
Static Group Object Class:	group	
Static Member DN Attribute:	member	
Static Group DNs from Member DN Filter:	(&(member=%M)(objectclass=group))	

13. Click **Save**.
14. On the Provider Summary page, reorder the providers in the following order, making sure that their Control Flags are set to `SUFFICIENT` where applicable:
    1. Negotiate Identity Asserter
    2. ActiveDirectoryAuthenticator (`SUFFICIENT`)
    3. DefaultAuthenticator (`SUFFICIENT`)
    4. Other authenticators...

33.5.3.3 Configuring WebCenter Portal

Once you have completed the steps for configuring the Negotiate Identity Assertion Provider and Active Directory Authenticator, and all applications on your WebLogic domain are configured for single sign-on with Microsoft clients in the required domain, a final step is required to provide a seamless single-sign-on experience for your users when accessing WebCenter Portal. There are two options for doing this:

- Turn off public access, by logging in to WebCenter Portal as an administrator and removing `View` access from the `Public-User` role. When public access is turned off, accessing the URL `http://host:port/webcenter` takes the user directly to the authenticated view rather than the default public page which has a login section. This is recommended when users are accessing WebCenter Portal only using Internet Explorer, and are confined to the domain where WNA is set up.
- If you must retain public access to WebCenter Portal, then the recommendation is to use the `oracle.webcenter.spaces.osso=true` flag when starting the `WC_Spaces` server. This flag tells WebCenter Portal that SSO is being used and no login form should be displayed on the default landing page. A Login link is displayed instead that the user can click to invoke the SSO authentication where the user will be automatically logged in. If Firefox is used to access WebCenter Portal within the Windows network configured for WNA, or any browser is used to access WebCenter Portal from outside the Windows network domain, users see the login page after clicking the Login link.

33.5.3.4 Configuring the Discussions Server for SSO

This section describes how to configure the discussions server for single sign-on. Before configuring the discussions server for SSO, ensure that it has been configured to use the same identity store LDAP as WebCenter Portal, as described in Section 31.4.1, "Migrating the Discussions Server to Use an External LDAP."

To set up the discussions server for SSO:

1. Log in to the Oracle WebCenter Portal's Discussion Server Server Admin Console at:

   `http://`*`host`*`:`*`port`*`/owc_discussions/admin`

   Where *`host`* and *`port`* are the host ID and port number of the `WC_Collaboration` managed server.

2. Open the System Properties page and edit (if it already exists) or add the `owc_discussions.sso.mode` property, setting it's value to `true`.

33.6 Configuring SSO with Virtual Hosts

This section describes the OHS configuration required for an environment containing applications that use "/" as the context root, and the additional configuration required in OHS when single sign-on is involved.

This section contains the following subsections:

- Section 33.6.1, "Understanding the Need for a Virtual Host"
- Section 33.6.2, "Configuring Virtual Hosts for OSSO"
- Section 33.6.3, "Configuring Virtual Hosts for OAM 10g"
- Section 33.6.4, "Configuring Virtual Hosts for OAM 11g"
- Section 33.6.5, "Configuring WebCenter Portal for Virtual Hosts"
- Section 33.6.6, "Testing Your Configuration"

33.6.1 Understanding the Need for a Virtual Host

The WebCenter Portal Suite includes a desktop integration application that uses "/" as the context root. If this application is to be used in a single sign-on environment you

need to route it through OHS. To do this without a virtual host we could add the following entry to `mod_wl_ohs.conf`:

```
<Location  />
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8888
</Location>
```

However, this would affect all context roots not explicitly defined, which brings us to the need for a virtual host.

The term *virtual host* refers to the practice of running more than one web site (such as `www.company1.com` and `www.company2.com`) on a single machine. Virtual hosts can be *IP-based*, meaning that you have a different IP address for each web site, or *name-based*, meaning that you have multiple names running on each IP address. The fact that they are running on the same physical server is not apparent to the end user. For more information about virtual hosts, refer to your Apache documentation.

33.6.2 Configuring Virtual Hosts for OSSO

This section describes the steps for configuring virtual hosts when OSSO is configured as the single sign-on solution. Prior to completing these steps you should already have completed the steps in Section 33.3, "Configuring Oracle Single Sign-On (OSSO)."

To use virtual hosts with OSSO you need to register partner applications with the virtual host option. Also, for `webtier-spaces.example.com`, you need to bypass single sign-on as some applications support only BASIC authentication and do not require single sign-on. These configurations are described in the following steps:

1. Move the `mod_osso.conf` file from `moduleconf` to the same location as `httpd.conf`. (All files in `moduleconf` are loaded automatically by default, but we need OSSO disabled for our virtual host.)
2. Update the virtual host setup in `httpd.conf` as shown in the following example:

```
NameVirtualHost *:7777

<VirtualHost *:7777>
  ServerName webtier.example.com
  include mod_osso.conf
</VirtualHost>

<VirtualHost *:7777>
  ServerName webtier-spaces.example.com
  <Location  />
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8888
  </Location>
  <Location /webcenter>
      Deny from all
  </Location>
  <Location /webcenterhelp>
      Deny from all
  </Location>
  <Location /rest>
      Deny from all
  </Location>
</VirtualHost>
```

By including the `mod_osso.conf` in the default virtual host we provide a single sign-on experience for the default virtual host (`webtier.example.com`), but not for the WebCenter Portal virtual host (`webtier-spaces.example.com`) as some applications do not support it.

3. Restart OHS. Also remember to update the DNS with entries for `webtier-spaces.example.com`.

> **Note:** In the `webtier-spaces.example.com` virtual host that bypasses single sign-on, only some applications need to bypass single sign-on. For other applications like WebCenter Portal, however, we need single sign-on so we deny access to these applications from this virtual host.

33.6.3 Configuring Virtual Hosts for OAM 10g

To configure OAM 10g for virtual hosts we need to bypass single sign-on for applications that only support BASIC authorization or do not require single sign-on. For more information, see the "Associating a WebGate with Particular Virtual Hosts, Directories, or Files"section in *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager with Oracle Security Token Service for 10g*.

Prior to completing these steps you should already have completed the steps for configuring OAM 10g in Section 33.2, "Configuring Oracle Access Manager (OAM)."

1. Locate and comment out the following configuration in `httpd.conf`:

```
#Comment out this and move to VirtualHost configuration
#<LocationMatch "/*">
#AuthType Oblix
#require valid-user
#</LocationMatch>
```

   This entry causes the WebGate to intercept all requests and process them.

2. Move this entry into the virtual host configuration where single sign-on is required as shown in the example below:

```
NameVirtualHost *:7777

<VirtualHost *:7777>
  ServerName webtier.example.com
  <LocationMatch "/*">
    AuthType Oblix
    require valid-user
  </LocationMatch>
</VirtualHost>

<VirtualHost *:7777>
  ServerName webtier-spaces.example.com
  <Location  />
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8888
  </Location>
  <Location /webcenter>
      Deny from all
  </Location>
  <Location /webcenterhelp>
      Deny from all
```

```
  </Location>
  <Location /rest>
      Deny from all
  </Location>
</VirtualHost>
```

The idea is to provide a single sign-on experience for the default virtual host (`webtier.example.com`), but not for the WebCenter Portal virtual host (`webtier-spaces.example.com`) as some applications do not support it.

3. Restart OHS. Also be sure to update the DNS with entries for `webtier-spaces.example.com`.

> **Note:** In the `webtier-spaces.example.com` virtual host that bypasses single sign-on, only some applications need to bypass single sign-on. For other applications like WebCenter Portal, however, we need single sign-on so we deny access to these applications from this virtual host.

33.6.4 Configuring Virtual Hosts for OAM 11g

To configure OAM 11g for virtual hosts we need to bypass single sign-on for applications that only support BASIC authorization or do not require single sign-on.

Prior to completing these steps you should already have completed the steps for configuring OAM 11g in Section 33.2, "Configuring Oracle Access Manager (OAM)."

Follow the steps below to configure virtual hosts for OAM 11g.

1. Locate and comment out the following configuration in `webgate.conf`:

```
#Comment out this and move to VirtualHost configuration
#<LocationMatch "/*">
#AuthType Oblix
#require valid-user
#</LocationMatch>
```

This entry causes the WebGate to intercept all requests and process it.

2. Move this entry into the virtual host configuration in `httpd.conf` where single sign-on is required. as shown in the example below:

```
NameVirtualHost *:7777

<VirtualHost *:7777>
  ServerName webtier.example.com
  <LocationMatch "/*">
    AuthType Oblix
    require valid-user
  </LocationMatch>
</VirtualHost>

<VirtualHost *:7777>
  ServerName webtier-spaces.example.com
  <Location  />
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8888
  </Location>
  <Location /webcenter>
```

```
    Deny from all
  </Location>
  <Location /webcenterhelp>
    Deny from all
  </Location>
  <Location /rest>
    Deny from all
  </Location>
</VirtualHost>
```

The idea is to provide a single sign-on experience for the default virtual host (`webtier.example.com`), but not for the WebCenter Portal virtual host (`webtier-spaces.example.com`) as some applications do not support it.

3. Restart OHS. Also be sure to update the DNS with entries for `webtier-spaces.example.com`.

> **Note:** In the `webtier-spaces.example.com` virtual host that bypasses single sign-on, only some applications need to bypass single sign-on. For other applications like WebCenter Portal, however, we need single sign-on so we deny access to these applications from this virtual host.

33.6.5 Configuring WebCenter Portal for Virtual Hosts

This section describes the additional configurations required for applications routed through the virtual host.

Sharepoint

Typically when you use the "Edit with Word" or similar features for MS Office products, the WebCenter Portal Sharepoint application obtains the host name and port name from the current request. However, in this case the Sharepoint application needs to be routed through the virtual host requiring that some system properties be set in `setDomainEnv` in the WebLogic domain. For a cluster setup, be sure to change these properties on every machine.

```
-Dnon_sso_host=webtier-spaces.example.com
-Dsso_base_url=http://webtier.example.com:7777
```

33.6.6 Testing Your Configuration

This section describes how you can test your virtual host and single sign-on configuration.

Sharepoint

1. Access `http://webtier.example.com:7777/webcenter` and check that you are challenged by SSO.
2. Log in and choose an MS Word document and click **Edit with Word**. Click **OK** when you see a confirmation dialog. Word should challenge you for BASIC authentication. Enter your credentials and you should be able to see the document
3. Navigate to **Office icon > Server > Document Management Information** and click **Open Site in Browser**. This should open the space to which the document belongs in your default browser.

Note that you will be prompted with a BASIC authentication challenge as MS Office integration has a restriction where it needs to go to the same URL as the one for the document. You will then be redirected to the space through `webtier.example.com` and be prompted for to login if not already logged i.

34

Configuring Portal Framework Applications for Single Sign-on

This chapter describes how to configure Portal Framework applications or Portlet Producer applications for single sign-on (SSO). All of the configurations described in this chapter assume that you have already configured SSO as described in Section 33, "Configuring Single Sign-on."

This chapter includes the following sections:

- Section 34.1, "Configuration Overview"
- Section 34.2, "Single Sign-on Prerequisites"
- Section 34.3, "Configuring the WebTier"
- Section 34.4, "Configuring Portal Framework and Portlet Producer Applications for OAM"
- Section 34.5, "Configuring Portal Framework Applications for OSSO"
- Section 34.6, "Configuring Portal Framework Applications for SAML SSO"

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console. Users with the `Monitor` or `Operator` roles can view security information but cannot make changes.

See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

34.1 Configuration Overview

Oracle Oracle WebCenter Portal supports single sign-on (SSO) for the following SSO solutions for Portal Framework applications:

- OAM 10g
- OAM 11g
- OSSO
- SAML SSO (Portal Framework application as a destination application)
- SAML SSO (Portal Framework application as the source application)

Before a Portal Framework application can participate in single sign-on, in addition to the SSo configuration described in Chapter 33, "Configuring Single Sign-on," you must

also configure the Portal Framework application itself for the chosen SSO solution. To do this, follow the instructions in Chapter 34.2, "Single Sign-on Prerequisites," and then continue with the steps for your particular solution. The only exception to this is for SAML SSO, where the Portal Framework application is acting as the source application where all the steps, including the prerequisites, are covered in Section 34.6.2, "Configuring SAML SSO for a Source Portal Framework Application."

34.2 Single Sign-on Prerequisites

All Portal Framework applications participating in SSO need to have certain common configurations in place regardless of the single sign-on solution used to protect the application. The only exception to this is for SAML SSO where the Portal Framework application is acting as the source application (see Section 34.6.2, "Configuring SAML SSO for a Source Portal Framework Application" for more information).

The common single sign-on prerequisites are covered in the following subsections:

- Section 34.2.1, "Adding CLIENT-CERT in web.xml"
- Section 34.2.2, "Setting the Cookie Path for JSESSIONID"
- Section 34.2.3, "Determining the Public and Protected URIs for Your Application"
- Section 34.2.4, "Implications of Embedded Login"
- Section 34.2.5, "Handling Logout"

34.2.1 Adding CLIENT-CERT in web.xml

All SSO solutions use an identity asserter configured on the WLS domain that asserts the type of assertion that an SSO configuration provides. For example, for OAM, it asserts based on the `ObSSOCookie` or `OAM_REMOTE_USER` header; for SAML SSO it asserts a SAML assertion.

For an asserter to assert identity, the application must specify `CLIENT-CERT` as its authentication method in its login configuration. Consequently, your application's `web.xml` file must have `CLIENT-CERT` specified as the `auth-method` as shown in the following example:

```
<login-config>
    <auth-method>CLIENT-CERT,FORM</auth-method>
```

Note that in Weblogic, you can specify comma-separated authentication methods. In this example, if the SSO assertion is not available (CLIENT-CERT), then the application will fall back to `FORM`-based authentication.

34.2.2 Setting the Cookie Path for JSESSIONID

For SSO setups, Oracle recommends that you set an application cookie path. You can do this in WLS by editing the `weblogic.xml` file and adding the following entry:

```
<session-descriptor>
  <cookie-path>/customportal</cookie-path>
</session-descriptor>
```

where `customportal` is the context root of your application.

34.2.3 Determining the Public and Protected URIs for Your Application

An SSO configuration involves specifying the public and protected URIs of your application. Some SSO solutions, like OSSO and SAML SSO, require only the protected URIs to be specified. The following list shows the typical protected and public URIs for a Portal Framework application:

Public URI:

```
/<app-context-root>
```

Protected URI:

```
/<app-context-root>/adfAuthentication
```

You can determine the protected URIs for your application by checking the `security-constraint` node of the `web.xml` file as shown in the following example:

```
<security-constraint>
  <web-resource-collection>
    <web-resource-name>adfAuthentication</web-resource-name>
    <url-pattern>/adfAuthentication</url-pattern>
  </web-resource-collection>
  <auth-constraint>
    <role-name>valid-users</role-name>
  </auth-constraint>
</security-constraint>
```

Note that the entries in the `security-constraint` node are always relative to the application context root. In this example, this security constraint translates to `/app-context-root/adfAuthentication`. If there were another security constraint specified, `/admin` for example, then that would translate to `/app-context-root/admin`.

34.2.4 Implications of Embedded Login

Portal Framework applications typically use a form-based login mechanism where a login page is configured in the `login-config` section of the `web.xml` configuration file (note that there is no separate login configuration file). Applications can also embed a login area in the page template, or a provide landing page. This usually submits the users credentials to `j_security_check` for authentication. For SSO, however, authentication must be done through an SSO login challenge.

34.2.5 Handling Logout

The ADF Authentication Servlet is equipped to handle logout for all SSO solutions, and your application's logout should invoke the ADF Authentication Servlet for logout. To do this, modify the navigation rule for successful logout in your application's `faces-config.xml` file as shown in the example below:

```
<navigation-case>
      <from-outcome>logout_success</from-outcome>
      <to-view-id>/adfAuthentication?logout=true&amp;end_url=/</to-view-id>
      <redirect/>
</navigation-case>
```

The `end_url` parameter for `/adfAuthentication` can be any URL that you want to direct the user to after a successful logout. For example, specifying `/` would take the user to the application's default page.

34.3 Configuring the WebTier

If your environment has a WebTier front-ending your enterprise applications you'll need to configure it for SSO. The WebTier is required for OAM and OSSO solutions, and is used in a SAML SSO solution when Content Server is involved.

1. Add a mapping for your application in `mod_wl_ohs.conf` as shown in the example below:

```
<Location /customportal>
      SetHandler weblogic-handler
      WebLogicHost webcenter.example.com
      WebLogicPort 8888
</Location>
```

   where `customportal` is the context root of your application. Or as in the following example:

```
<IfModule mod_weblogic.c>
MatchExpression /webcenter
WebLogicHost=<Host_Name>|WebLogicPort=<Port>
</IfModule>
```

   where `<Host_Name>` and `<Port>` are the host ID and port number of the host server. For example:

```
<IfModule mod_weblogic.c>
MatchExpression /webcenter WebLogicHost=example.com|WebLogicPort=12345
</IfModule>
```

2. Restart the Oracle HTTP Server.

34.4 Configuring Portal Framework and Portlet Producer Applications for OAM

This section describes how to configure your Framework or Portlet Producer application for OAM 10*g* and 11*g*. Prior to following the steps in this section you should already have followed the instructions in Section 33.2, "Configuring Oracle Access Manager (OAM)" to set up SSO for WebCenter Portal and related applications. You should also have completed the configurations in Section 34.2, "Single Sign-on Prerequisites."

> **Note:** Prior to starting, you should already have configured the required OAM Asserter and Authenticator pointing to the identity store LDAP used by OAM in the domain where your Portal Framework application is deployed. If you have not done this, follow the instructions in Section 33.2, "Configuring Oracle Access Manager (OAM)" before starting.

This section includes the following subsections:

- Section 34.4.1, "Configuring Portal Framework Applications for OAM 10g"
- Section 34.4.2, "Configuring Portlet Producer Applications for OAM 10g"
- Section 34.4.3, "Configuring Portal Framework Applications for OAM 11g"
- Section 34.4.3, "Configuring Portal Framework Applications for OAM 11g"

34.4.1 Configuring Portal Framework Applications for OAM 10*g*

This section describes how to configure a Portal Framework application for single sign-on using OAM 10*g*. Prior to configuring your application you should already have completed the OAM installation and configuration as described in Section 33.2, "Configuring Oracle Access Manager (OAM)."

To configure a Portal Framework application for OAM 10*g*:

1. Log in to the OAM Console using your browser to navigate to:

   `http://host:port/access/oblix`

2. Click **Policy Manager**.
3. Locate the policy domain that you created to protect your WebCenter Portal resources.
4. Open the Resources tab and click **Add**.
5. Add the resources. For each resource:
   a. Select `HTTP` as the **Resource Type**.
   b. Select the **Host Identifier** for the WebCenter Portal Web Tier.
   c. Enter the **URL Prefix** (`/<app-context-root>`) for the application.
   d. Enter a **Description** for the resource.
   e. Make sure that **Update Cache** is selected, and then click **Save**.
6. Repeat step 5 to add `/<app-context-root>/adfAuthentication` as a resource.
7. Go to the Policies tab and locate the public policy.
8. Open the policy and select the resource created in step 5 (i.e., `/<app-context-root>`).
9. Save your changes.
10. Restart the WebTier and test your changes.

34.4.2 Configuring Portlet Producer Applications for OAM 10*g*

This section describes how to configure Portlet Producer applications for single sign-on using OAM 10*g*. Prior to configuring your Portlet Producer application follow the steps in Section 34.4.1, "Configuring Portal Framework Applications for OAM 10g," then complete the steps below.

To configure a Portlet Producer application for OAM 10*g*:

1. Log in to the OAM Console using your browser to navigate to:

   `http://host:port/access/oblix`

2. Click **Policy Manager**.
3. Locate the policy domain that you created to protect your WebCenter Portal resources.
4. Open the Resources tab and click **Add**.
5. Select `HTTP` as the **Resource Type**.
6. Select the **Host Identifier** for the WebCenter Portal WebTier.
7. Enter the **URL Prefix** (`/<app-context-root/portlets>`) for the application.

8. Enter a **Description** for the resource.
9. Make sure that **Update Cache** is selected, and then click **Save**.
10. Go to the Policies tab and locate the Exclusion Scheme policy and select the newly created Portlet Producer resource for this policy.
11. Open the policy and select the resource created in step 5 (i.e., `/<app-context-root>/portlets`).
12. Save your changes.
13. Open the Resources tab and click **Add**.
14. Select `HTTP` as the **Resource Type**.
15. Select the **Host Identifier** for the WebCenter Portal WebTier.
16. Enter the **URL Prefix** (`/<app-context-root/monitor>`) for the application.
17. Enter a **Description** for the resource.
18. Make sure that **Update Cache** is selected, and then click **Save**.
19. Save your changes. By default they will be included under protected policy.
20. Restart the WebTier and test your changes.

34.4.3 Configuring Portal Framework Applications for OAM 11*g*

This section describes how to configure a Portal Framework application for single sign-on using OAM 11*g*. Prior to configuring your application you should already have completed the OAM installation and configuration as described in Section 33.2, "Configuring Oracle Access Manager (OAM)."

To configure a Portal Framework application for OAM 11*g*:

1. Log in to the OAM Console using your browser to navigate to:

   `http://host:port/oamconsole`

2. Go to **Policy Configuration > Application Domains**.

   The Policy Manager pane displays.

3. Locate the application domain you created using the name used when registering the WebGate agent.
4. Open the Resources tab and click **New Resource**.
5. Add the resources for the Portal Framework application. For each resource:
   a. Select `HTTP` as the **Resource Type**.
   b. Select the **Host Identifier** created while registering the WebGate agent.
   c. Enter the **Resource URL** (`/<app-context-root>*`) for the application.
   d. Enter a **Description** for the resource.
   e. Set the **Protection Level** to `Unprotected`.
   f. Set the **Authentication Policy** to `Public Resource Policy`.
   g. Set the **Authorization Policy** to `Protected Resource Policy`.
   h. Click **Apply**.
6. Repeat step 5 to add `/<app-context-root>/.../*` as a resource.

7. Add `/<app-context-root>/adfAuthentication*` as a resource:
   a. Select `HTTP` as the **Resource Type**.
   b. Select the **Host Identifier** created while registering the WebGate agent.
   c. Enter the **Resource URL** (`/<app-context-root>/adfAuthentication*`) for the application.
   d. Enter a **Description** for the resource.
   e. Set the **Protection Level** to `Protected`.
   f. Set the **Authentication Policy** to `Protected Resource Policy`.
   g. Set the **Authorization Policy** to `Protected Resource Policy`.
   h. Click **Apply**.
8. Add `/<app-contextroot>/monitor*` as a resource:
   a. Select `HTTP` as the **Resource Type**.
   b. Select the **Host Identifier** created while registering the WebGate agent.
   c. Enter the **Resource URL** (`/<app-contextroot>/monitor*`) for the application.
   d. Enter a **Description** for the resource.
   e. Set the **Protection Level** to `Protected`.
   f. Set the **Authentication Policy** to `Protected Resource Policy`.
   g. Set the **Authorization Policy** to `Protected Resource Policy`.
   h. Click **Apply**.
9. Add `/<app-contextroot>/monitor/.../*` as a resource:
   a. Select `HTTP` as the **Resource Type**.
   b. Select the **Host Identifier** created while registering the WebGate agent.
   c. Enter the **Resource URL** (`/<app-contextroot>/monitor/.../*`) for the application.
   d. Enter a **Description** for the resource.
   e. Set the **Protection Level** to `Protected`.
   f. Set the **Authentication Policy** to `Protected Resource Policy`.
   g. Set the **Authorization Policy** to `Protected Resource Policy`.
   h. Click **Apply**.
10. Restart the WebTier and test your changes.

34.4.4 Configuring Portlet Producer Applications for OAM 11*g*

This section describes how to configure Portlet Producer applications for single sign-on using OAM 11*g*. Prior to configuring your Portlet Producer application follow the steps in Section 34.4.3, "Configuring Portal Framework Applications for OAM 11g," then complete the steps below.

To configure a Portlet Producer application for OAM 11*g*:

1. Log in to the OAM Console using your browser to navigate to:

```
http://host:port/access/oblix
```

2. Click **Policy Manager**.
3. Locate the policy domain that you created to protect your WebCenter Portal resources.
4. Open the Resources tab and click **Add**.
5. Select `HTTP` as the **Resource Type**.
6. Select the **Host Identifier** created while registering the WebGate agent.
7. Enter the **URL Prefix** (`/<app-contextroot>/portlets/.../*`) for the application.
8. Enter a **Description** for the resource.
9. Set the **Protection Level** to `Excluded`, and then click **Save**.
10. Restart the WebTier and test your changes.

34.5 Configuring Portal Framework Applications for OSSO

This section describes how to configure your Portal Framework application for OSSO. Prior to following the steps in this section you should already have followed the instructions in Section 33.3, "Configuring Oracle Single Sign-On (OSSO)" to set up SSO for WebCenter Portal and related applications. You should also have completed the configurations in Section 34.2, "Single Sign-on Prerequisites."

> **Note:** Prior to starting, you should already have configured the required OSSO Asserter and Authenticator pointing to the identity store (OID) used by OSSO in the domain where your Portal Framework application is deployed. If you have not done this, follow the instructions Section 33.3, "Configuring Oracle Single Sign-On (OSSO)" before starting.

To configure a Portal Framework application for OSSO:

1. Locate and open the `mod_osso.conf` file in OHS.
2. Add the following entry for your Portal Framework application to the other similar entries:

```
<Location /<app-context-root>/adfAuthentication>
      OssoSendCacheHeaders off
      require valid-user
      AuthType Osso
</Location>
```

3. Restart OHS.

34.6 Configuring Portal Framework Applications for SAML SSO

This section describes how to set up SAML SSO for Portal Framework applications. Note that SAML single sign-on is only recommended for smaller environments (a department, for example) where no enterprise SSO solution is available.

The steps are divided into two scenarios:

- **Scenario 1: A Portal Framework application as a destination application**

This is the default SAML SSO behavior provided by the WebCenter Portal SAML SSO scripts, where WebCenter Portal is the source application and all other applications are destination applications (that is, you need to be logged into WebCenter Portal for single sign-on with other destination applications to work).

- **Scenario 2: A Portal Framework application as a source application**

  This behavior is not supported by the WebCenter Portal SAML SSO scripts and requires manual configuration. In this instance you want your Portal Framework application to act as the SAML source, and other applications (including WebCenter Portal) to act as destination applications (that is, your Portal Framework application is the first point of access and you need to be logged into it for single sign-on with other destination applications to work.

These two approaches to configuring SAML for Portal Framework applications is described in the following subsections:

- Section 34.6.1, "Configuring SAML SSO for a Destination Portal Framework Application"
- Section 34.6.2, "Configuring SAML SSO for a Source Portal Framework Application"

34.6.1 Configuring SAML SSO for a Destination Portal Framework Application

Prior to following the steps in this section, you should already have completed the prerequisites and steps in Section 33.4, "Configuring SAML-based Single Sign-on" that describe how to set up SSO for WebCenter Portal and related applications. The steps in this section supplement that setup with configuration steps for your Portal Framework application.

In this scenario, WebCenter Portal continues to act as the source application with your new Portal Framework application participating in single sign-on as a destination application (that is, if you are logged into WebCenter Portal, when you access one of your Portal Framework application's protected URIs, you are automatically logged in). If you are not already logged into WebCenter Portal and you access a protected URI, you will be directed to the WebCenter Portal login page and then redirected back to your application's secure page.

The steps below assume that:

- Your Portal Framework application is deployed in the WebCenter Portal domain where the `configureSpaces.py` script was run. If your Portal Framework application is in a different domain, then you'll need to create a `SAMLIdentityAsserterV2` ID Asserter in the WLS Administration console (**security realm > providers > authenticator**) and restart WLS. You then need to export the certificate used in your SAML SSO setup and register it under the SAML identity asserter you created.
- The steps and example parameter values below assume you are using the `demoidentity` certificate. If you are using a different certificate, change the certificate name where appropriate.
- If a WebTier is part of the configuration, the host and port IDs are those of the WebTier host and WebTier port.

This section includes the following subsections:

- Section 34.6.1.1, "Enabling the Destination Site"
- Section 34.6.1.2, "Configuring a Relying Party"

- Section 34.6.1.3, "Configuring an Asserting Party"

34.6.1.1 Enabling the Destination Site

To enable the destination site for your Portal Framework application:

1. Log onto the WLS Administration Console for the WebCenter Portal domain.
2. Select **Servers > [ServerHostingPortalApp] > Configuration > Federation Services > SAML 1.1 Destination Site**.
3. Enter the parameters for the destination site as shown in Table 34–1:

Table 34–1 Destination Site Parameters

Parameter	Value	Description
Destination Site Enabled	Selected	Specifies whether the destination site is enabled.
ACS Requires SSL	Unselected	Specifies whether the Assertion Consumer Service requires SSL. If checked, then ensure that ACS URL specified in Credential Mapper's relying party uses HTTPS and target server's SSL port.
Assertion Consumer URIs	`/<app-context-root>/samlacs/acs` (add on top, leave the rest as is)	The Assertion Consumer URIs. In this case, we have chosen for the ACS to reside within the target application so that it uses the same login cookie.
POST Recipient Check Enabled	Selected	Specifies whether the POST recipient check is enabled. When true, the recipient of the SAML Response must match the URL in the HTTP Request.
POST One use Check Enabled	Selected	Specifies whether the POST one-use check is enabled.

4. Save your changes, leaving the rest as their default values.
5. Restart the server hosting the Portal Framework application.

34.6.1.2 Configuring a Relying Party

To configure a relying party for your Portal Framework application:

1. Log onto the WLS Administration Console for the WebCenter Portal domain.
2. Select **Security Realms > Providers > Credential Mapping > wcsamlcm > Management > Relying Parties**.
3. Create a new relying party using the parameters in Table 34–2:

Table 34–2 Relying Party Parameters

Parameter	Value	Description
Profile	`Browser/POST`	The SAML profile used by this SAML Relying Party.
Enabled	Selected	The state of this SAML Relying Party.
Description	Portal Framework application	A short description of this Relying Party

Table 34–2 (Cont.) Relying Party Parameters

Parameter	Value	Description
Target URL	`http://host:port/<app-context-root>`	The destination site URL for which authentication is requested.
Assertion Consumer URL	`http://host:port/<app-context-root>/samlacs/acs`	The URL at which an Assertion Consumer Service for this SAML Relying Party can be reached. Indicates the URL to which an assertion or artifact should be POSTed or redirected. **Note:** If you have checked ACS requires SSL while configuring destination site federation services, then use https protocol and the SSL port for the managed server
Assertion Consumer Properties	`APID=ap_0000X`	The **X** points to ID of the asserting party you will create in next step.
Sign Assertions	Selected	Specifies whether generated assertions for this SAML Relying Party are signed.
Include KeyInfo	Selected	Indicates whether a <ds:keyinfo> element containing the signing certificate should be included when signing assertions. Default value is true. This value is ignored if Sign Assertions is false.

4. Save your changes, leaving the rest as their default values.

34.6.1.3 Configuring an Asserting Party

To configure an asserting party for your Portal Framework application:

1. Log onto the WLS Administration Console for the WebCenter Portal domain.
2. Select **Security Realms > Providers > Authentication > wcsamlia > Management > Asserting Parties**.
3. Create a new asserting party using the parameters in Table 34–3:

Table 34–3 Asserting Party Parameters

Parameter	Value	Description
Profile	`Browser/POST`	The SAML profile used with this partner.
Enabled	Selected	Specifies whether this Asserting Party can be used to obtain SAML assertions
Description	WebCenter Portal	A short description of this Asserting Party
Target URL	`http://host:port/webcenter`	The target URL of this SAML asserting party.
POST Signing Certificate alias	`demoidentity`	The alias of the certificate trusted for verifying signatures on SAML protocol elements from this asserting party. Must be set for Browser/POST profile.

Table 34–3 (Cont.) Asserting Party Parameters

Parameter	Value	Description
Source Site Redirect URIs	`/<app-context-root>/adfAuthentication`	An optional set of URIs from which unauthenticated users will be redirected to the configured ITS URL. If set, the IntersiteTransferURL must also be set. **Note:** Due to this setting, when you access the destination site first, you are redirected to the ITS url configured which in this case is within the source app, your session is established source app and then redirected to the destination site.
Source Site ITS URL	`http://host:port/webcenter/samlits/its`	The Intersite Transfer Service (ITS) URL of the SAML Source Site for this asserting party. Use this with SSO profiles only, to support the destination site as the first access point scenario, whereby a user trying to access a destination site URL prior to being authenticated is redirected to the source site to be authenticated and obtain a SAML assertion. The Redirect URIs attribute must also be configured for source-site redirection to work. **Note:** If you check ITS requires SSL in Source Site Federation Services, then you need to change Source Site ITS URL to use HTTPS and the server's SSL port.
Source Site ITS parameters	`RPID=rp_0000x`	Replace the **x** with the ID of the relying party you created previously.
Issuer URI	`http://www.oracle.com/webcenter`	The issuer URI of the SAML Authority issuing assertions for this SAML asserting party.
Signature Required	Selected	If true, assertions must be signed. If false, signature elements are not required, but will be verified if present.
Assertion Signing Certificate alias	`demoidentity`	

4. Save your changes, leaving the rest as their default values.
5. Continue by testing that single sign-on works as expected.

34.6.2 Configuring SAML SSO for a Source Portal Framework Application

In this scenario the Portal Framework application acts as the source application and other applications (like WebCenter Portal) are the destinations. For configurations that include Content Server, prior to completing the configurations in this section you should have followed the relevant steps in Section 33.4.2.1.1, "Configuring Oracle Content Server for SAML SSO."

The steps below are based on the following assumptions:

- The WebCenter Portal SAML SSO scripts have not been run. The scripts configure WebCenter Portal to act as the source application, so these steps should be done manually.
- You are using the default `demoidentity` certificate and you have already exported the certificate from the domain hosting your Portal Framework application into `demoidentity.der`.
- Your Portal Framework application is `/customportal`.
- For configurations that include Content Server and if a WebTier is part of the configuration, the host and port IDs are those of the WebTier host and WebTier port.

This section contains the following subsections:

34.6.2.1 Protecting SAML ITS

In the `web.xml` file of your Portal Framework application, add the following entry after the entry for protecting `/adfAuthentication`:

```
<security-constraint>
    <web-resource-collection>
      <web-resource-name>samlits</web-resource-name>
      <url-pattern>/samlits/its</url-pattern>
    </web-resource-collection>
    <auth-constraint>
      <role-name>valid-users</role-name>
    </auth-constraint>
  </security-constraint>
```

34.6.2.2 Setting the Cookie Path for JSESSIONID

For SSO setups, Oracle recommends that you set a cookie path to the context root of your application. You can do this in WLS by editing the `weblogic.xml` file and adding the following entry:

```
<session-descriptor>
    <cookie-path>/customportal</cookie-path>
  </session-descriptor>
```

where `customportal` is the context root of your application.

34.6.2.3 Setting the SSO Property to True

Since the WebCenter Portal application now acts as a destination application, you need to hide the login area on the WebCenter Portal landing page. To do this, set the

following property in your `setDomainEnv` file and restart the `WC_Spaces` server for the changes to take effect.

```
EXTRA_JAVA_PROPERTIES="-Doracle.webcenter.spaces.osso=true
${EXTRA_JAVA_PROPERTIES}"
export EXTRA_JAVA_PROPERTIES
```

34.6.2.4 Configuring the SAML Credential Mapping Provider

In the security realm of the domain hosting your Portal Framework application, create a SAML Credential Provider V2 (SAMLCredentialMapperV2) instance. Note that the SAML Credential Mapping provider is not part of the default security realm. Configure the SAML Credential Mapping provider as a SAML authority, using the **Issuer URI**, **Name Qualifier**, and other attributes as shown in Table 34–4:

Table 34–4 SAML Credential Mapping Provider Parameters

Parameter	Value	Description
Issuer URI	`http://www.oracle.com/webcenter`	The Issuer URI (name) of this SAML Authority. This unique URI tells the destination site (`owc_wiki`) the origin of the SAML message and allows it to match it with the key. Typically, the URL is used to guarantee uniqueness.
Name Qualifier	`oracle.com`	The Name Qualifier value used by the Name Mapper. The value of the Name Qualifier is the security or administrative domain that qualifies the name of the subject. This provides a means to federate names from disparate user stores while avoiding the possibility of subject name collisions.
Signing Key Alias	`demoidentity`	The alias used to retrieve from the keystore the key that is used to sign assertions.
Signing Key Passphrase	`DemoIdentityPassPhrase`	The credential (password) used to retrieve from the keystore the keys used to sign assertions.

Save your changes, accepting the defaults for the rest of the parameters, and restart all of WLS.

34.6.2.5 Configuring a Relying Party

You'll need to configure relying parties for each of the destination applications. The steps below show you how to do using WebCenter Portal as an example. For other applications, refer to Table 34–10 in Section 34.6.2.9, "Configuring Other Destination Applications," and modify the highlighted values appropriately using the steps below as a reference.

To configure a relying party for WebCenter Portal:

1. Log onto the WLS Administration Console for the WebCenter Portal domain.
2. Select **Security Realms > RealmName > Providers > Credential Mapping > SAMLCredentialMapperName > Management > Relying Parties**.
3. Create a new relying party using the parameters in Table 34–5:

Table 34–5 WebCenter Portal Parameters for Relying Party

Parameter	Value	Description
Profile	`Browser/POST`	The SAML profile used by this SAML Relying Party.
Enabled	Selected	The state of this SAML Relying Party.
Description	WebCenter Portal	A short description of this Relying Party
Target URL	`http://host:port/webcenter`	The destination site URL for which authentication is requested.
Assertion Consumer URL	`http://host:port/webcenter/samlacs/acs`	The URL at which an Assertion Consumer Service for this SAML Relying Party can be reached. Indicates the URL to which an assertion or artifact should be POSTed or redirected. **Note:** If you have checked ACS requires SSL while configuring destination site federation services, then use https protocol and the SSL port for the managed server
Assertion Consumer Properties	`APID=ap_00001`	One or more optional query parameters, in the form name=value, that will be added to the ACS URL when redirecting to the destination site. For a POST profile, these parameters will be included as form variables when using the default POST form. In this case, `ap_00001` indicates the ID of the asserting party for your Portal Framework application (`customportal`), which we will configure later in the SAML Identity Asserter of the domain hosting the Portal Framework application, and which provides the source site and ITS details.
Sign Assertions	Selected	Specifies whether generated assertions for this SAML Relying Party are signed.
Include KeyInfo	Selected	Indicates whether a <ds:keyinfo> element containing the signing certificate should be included when signing assertions. Default value is true. This value is ignored if Sign Assertions is false.

4. Save your changes, leaving the rest as their default values.

34.6.2.6 Configuring the Source Site Federation Services

To configure the source site Federation:

1. Log onto the WLS Administration Console for the WebCenter Portal domain.
2. Select **Environment > Servers >** *ServerHostingCustomPortal* **> Configuration > Federation Services >SAML 1.1 Source Site**.

3. Configure the SAML source site attributes as shown in Table 34–6.

Table 34–6 Source Site Federation Services Parameters

Parameter	Value	Description
Source Site Enabled	Selected	Allow the WebLogic server instance to serve as a SAML source site by setting Source Site Enabled to true.
Source Site URL	`http://host:port/customportal`	Set the URL for the SAML source site. This is the URL that hosts the Intersite Transfer Service and Assertion Retrieval Service. The source site URL is encoded as a source ID in hex and Base64.
Signing Key Alias	demoidentity	The SAML source site requires a trusted certificate with which to sign assertions. Add this certificate to the keystore and enter the credentials (alias and passphrase) to be used to access the certificate. The server's SSL identity key/certificates will be used by default if a signing alias and passphrase are not supplied.
Signing Key Passphrase	`DemoIdentityPassPhrase`	The SAML source site requires a trusted certificate with which to sign assertions. Add this certificate to the keystore and enter the credentials (alias and passphrase) to be used to access the certificate. The server's SSL identity key/certificates will be used by default if a signing alias and passphrase are not supplied.
Intersite Transfer URIs	`/customportal/samlits/its` (Add on top, leave the rest as is.)	Specify the URIs for the Intersite Transfer Service and (to support Browser/Artifact profile) the Assertion Retrieval Service. These URIs are also specified in the configuration of an Asserting Party.
Assertion Retrieval URIs	`/customportal/samars/ars` (Add on top, leave the rest as is.)	Applicable only when Artifact profile is used for REST.
ITS Requires SSL	Deselected	If you select this, then you need to change the Source Site ITS URL specified in the SAML Asserting party configuration in SAML Identity provider as HTTPS and the server's SSL port.
ARS Requires SSL	Deselected	Applicable only when Artifact profile is used

4. Save your changes, leaving the rest at their default values.
5. Restart the server hosting the Portal Framework application.

34.6.2.7 Configuring the SAML Identity Assertion Provider

To configure the SAML identity assertion provider:

1. Log onto the WLS Administration Console for the WebCenter Portal domain.
2. Create a SAML Identity Assertion Provider V2 instance as described in Section 34.6.1.3, "Configuring an Asserting Party," restarting all of WLS after saving your changes.
3. Log back onto the WLS Administration Console and go to **Security Realms > *RealmName* > Providers > Authentication > *`SAMLIdentityAsserterName`* > Management > Certificates**.
4. Configure a certificate for the SAML identity asserter:
5. Configure a certificate for the SAML identity asserter using the values shown in Table 34–7.

Table 34–7 Identity Asserter Certificate Parameters

Parameter	Value	Description
alias	`demoidentity`	Name you would you like to assign to your new certificate.
Path	`WEBLOGIC_HOME/server/lib/demoidentity.der`	Specify the path name of a .pem or .der file containing the X509 certificate you wish to import.

6. Go to **Security Realms > *RealmName* > Providers > Authentication > *`SAMLIdentityAsserterName`* > Management > Asserting Parties**.
7. Create a new asserting party using the parameters in Table 34–8. Use the same profile you chose for the corresponding relying party in Section 34.6.2.5, "Configuring a Relying Party."

Table 34–8 Asserting Parties Parameters

Parameter	Value	Description
Profile	`Browser/POST`	The SAML profile used with this partner.
Enabled	Selected	Specifies whether this Asserting Party can be used to obtain SAML assertions
Description	Portal Framework application for WebCenter Portal	A short description of this Asserting Party
Target URL	`http://host:port/customportal`	The target URL of this SAML asserting party.
POST Signing Certificate alias	`demoidentity`	The alias of the certificate trusted for verifying signatures on SAML protocol elements from this asserting party. Must be set for Browser/POST profile.

Table 34–8 (Cont.) Asserting Parties Parameters

Parameter	Value	Description
Source Site Redirect URIs	`/webcenter/adfAuthentication`	An optional set of URIs from which unauthenticated users will be redirected to the configured ITS URL. If set, the IntersiteTransferURL must also be set. **Note:** This setting, when you access the destination site first, redirects you to the ITS URL configured (which in this case is within the source application), your session is established for the source application, and you are then redirected to the destination site.
Source Site ITS URL	`http://host:port/customportal/samlits/its T`	The Intersite Transfer Service (ITS) URL of the SAML Source Site for this asserting party. Use this with SSO profiles only, to support the destination site as the first access point scenario, whereby a user trying to access a destination site URL prior to being authenticated is redirected to the source site to be authenticated and obtain a SAML assertion. The Redirect URIs attribute must also be configured for source-site redirection to work. **Note:** If you check ITS requires SSL in Source Site Federation Services, then you need to change Source Site ITS URL to use HTTPS and the server's SSL port.
Source Site ITS parameters	`RPID=rp_00001`	Optionally, zero or more query parameters, of the form name=value, that will be added to the ITS URL when redirecting to the source site. In this case rp_00001 is the relying party ID for WebCenter Portal as specified in the SAML Credential Mapping provider of the WLS domain for the Portal Framework application that provides the destination site details.
Issuer URI	`http://www.oracle.com/webcenter`	The issuer URI of the SAML Authority issuing assertions for this SAML asserting party.
Signature Required	Selected	If true, assertions must be signed. If false, signature elements are not required, but will be verified if present.
Assertion Signing Certificate alias	`demoidentity`	The alias of the certificate trusted for verifying signatures on assertions from this Asserting Party. This must be set if Signature Required is true. The certificate must also be registered in the SAML Identity Asserter's certificate registry.

8. Save your changes, leaving the rest at their default values.

34.6.2.8 Configuring the Destination Site Federation Services

To configure the destination site federation services:

1. From the WLS Administration Console, go to **WC Domain > WC_Spaces > Configuration > Federation Services > SAML 1.1 Destination Site [Spaces]**
2. Configure the SAML destination site attributes using the values in Table 34–9.

Table 34–9 Destination Site Parameters

Parameter	Value	Description
Destination Site Enabled	Selected	Specifies whether the Destination Site is enabled.
ACS Requires SSL	Deselected	Specifies whether the Assertion Consumer Service requires SSL. If checked, then ensure that the ACS URL specified in the Credential Mapper's relying party uses HTTPS and the target server's SSL port.

Assertion Consumer URIs	`/webcenter/samlacs/acs`
	`/rss/samlacs/acs`
	`/rest/samlacs/acs`
	(add on top, leave rest as is)
		The Assertion Consumer URIs. In this case, we have chosen for the ACS to reside within the target application so that it uses the same login cookie.

POST Recipient Check Enabled	Selected	Specifies whether the POST recipient check is enabled. When true, the recipient of the SAML Response must match the URL in the HTTP Request.
POST One use Check Enabled	Selected	Specifies whether the POST one-use check is enabled.

3. Save your changes, leaving the rest at their default values.
4. Restart the WebCenter Portal server.

34.6.2.9 Configuring Other Destination Applications

If you want applications other than WebCenter Portal to act as destination applications for your Portal Framework application, then perform the following steps:

1. Ensure you have the SAML ID asserter and certificate registered in each domain that hosts destination applications (refer to steps 1 - 5 of section Section 34.6.2.7, "Configuring the SAML Identity Assertion Provider").
2. Create a relying party for your destination application in the WLS domain hosting your Portal Framework application as you did for WebCenter Portal in Section 34.6.2.5, "Configuring a Relying Party." See Table 34–10 for appropriate values for each application.
3. In your destination application's WLS domain, create a corresponding asserting party similar to what you did for WebCenter Portal. Use the steps for creating an asserting party in Section 34.6.2.7, "Configuring the SAML Identity Assertion Provider." Be sure to set the source redirect URI appropriately to the secure URI for your destination application. See Table 34–10 for appropriate values for each application.

4. Ensure your asserting and relying parties are enabled and point to each other appropriately. That is, the Source Site ITS parameters in the asserting party and the Assertion Consumer Properties in the relying party point to each other appropriately.
5. Ensure you have enabled destination site federation services for the server hosting your destination application, and have added entries for `/yourdestinationapp/samlacs/acs` similar to what you did for the `WC_Spaces` server in as you did in Section 34.6.2.8, "Configuring the Destination Site Federation Services." See Table 34–10 for appropriate values for each application.

Table 34–10 Settings for Destination Applications Other than WebCenter Portal

Destination Application	Target URL (Relying Party)	ACS URL (Relying Party)	ACS URI (DestinationSiteFederationServices)	Source Redirect URI (Asserting Party)
RSS	http://host:port/rss	http://host:port/rss/samlacs/acs	/rss/samlacs/acs	/rss/rssservlet
REST	http://host:port/rest	http://host:port/rest/samlacs/acs	/rest/samlacs/acs	/rest/api/resourceIndex
Discussions	http://host:port/owc_discussions	http://host:port/owc_discussions/samlacs/acs	/owc_discussions/samlacs/acs	/owc_discussions/admin/forum-main.jsp /owc_discussions/admin/content-main.jsp /owc_discussions/login!withRedirect.jspa /owc_discussions/login!default.jspa /owc_discussions/login.jspa
ActivityGraph Engines	http://host:port/activitygraph-engines	http://host:port/activitygraph-engines/samlacs/acs	/activitygraph-engines/samlacs/acs	/activitygraph-engines/index.jsp
Content Server	http://host:port	http://host:port/samlacs/acs	/samlacs/acs	/adfAuthentication
Worklist Detail	http://host:port/workflow/WebCenterWorklistDetail	http://host:port/workflow/WebCenterWorklistDetail/samlacs/acs	/WebCenterWorklistDetail/samlacs/acs	/workflow/WebCenterWorklistDetail/faces/adf.task-flow
Worklist SDP	http://host:port/workflow/sdpmessagingsca-ui-worklist	http://host:port/workflow/sdpmessagingsca-ui-worklist/samlacs/acs	/sdpmessagingsca-ui-worklist/samlacs/acs	/workflow/sdpmessagingsca-ui-worklist/faces/adf.task-flow
Worklist Integration	http://host:port/integration/worklistapp	http://host:port/integration/worklistapp/samlacs/acs	/worklistapp/samlacs/acs	/integration/worklistapp/ssologin /integration/worklistapp/faces/home.jspx

35

Configuring SSL

This chapter describes how to secure WebCenter Portal and Portal Framework applications and components with SSL.

This chapter includes the following sections:

Note: The following can use WS-Security with message protection, and consequently have no hard requirement for SSL:

- BPEL servers - Worklist
- WSRP Producers
- Microsoft Live Communication Server (LCS) - Instant messaging and presence
- Discussions and announcements

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console. Users with the `Monitor` or `Operator` roles can view security information but cannot make changes.

See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

35.1 Securing the Browser Connection to WebCenter Portal with SSL

Securing the browser connection to WebCenter Portal with SSL consists of the following steps:

- Section 35.1.1, "Creating the Custom Keystore"
- Section 35.1.2, "Configuring the Custom Identity and Java Trust Keystores"
- Section 35.1.3, "Configuring the SSL Connection"

35.1.1 Creating the Custom Keystore

The first step is to generate a custom keystore for WebCenter Portal.

To create a custom keystore:

1. Go to *JDK_HOME*`/bin/` and open a command prompt.
2. Using keytool, generate a key pair:

   ```
   keytool -genkeypair -keyalg RSA -dname "dname" -alias alias -keypass
   key_password -keystore keystore -storepass keystore_password -validity
   days_valid
   ```

   Where:

   - *dname* is the DN (distinguished name) to use (for example, `cn=customidentity,dc=example,dc=com`)
   - *alias* is the alias to use (for example, `webcenter_wls`)
   - *key_password* is the password for the new public key, (for example, `MyPassword1`)
   - *keystore* is the keystore name, (for example, `webcenter_wls.jks`)
   - *keystore_password* is the keystore password, (for example, `MyPassword1`)
   - *days_valid* is the number of days for which the key password is valid (for example, `360`).

   Note: You must use the `-keyalg` parameter and specify `RSA` as its value as shown above as the default algorithm (DSA) used by `keytool` for generating the key is incompatible with Oracle WebServices Security Manager requirements.

3. Export the certificate containing the public key so WebCenter Portal clients can import it into their trust store:

```
keytool -exportcert -v -alias alias -keystore keystore
-storepass keystore_password -rfc -file certificate_file
```

Where:

- *alias* is the WebCenter Portal alias (for example, webcenter_wls)
- *keystore* is the keystore name, (for example, webcenter_wls.jks)
- *keystore_password* is the keystore password, (for example, MyPassword1)
- *certificate_file* is the file name for the certificate to export the key to (for example, webcenter_wls.cer)

4. Determine the trust store to use:

   Since you are using a self-signed certificate, you must update it as a trusted certificate in the server trust store. To do this, you must determine your trust store by going to the server:

   a. Log into the WebLogic Server Administration Console.

      For information on logging into the WebLogic Server Administration Console, see Section 1.13.2, "Oracle WebLogic Server Administration Console."

   b. In the Domain Structure pane, expand Environments and click Servers.

   c. In the list of servers, click WC_Spaces.

   d. Open the Configuration tab, and the Keystores subtab.

      The Keystores Settings pane displays.

   e. Note down the location of the server in the **Java Standard Trust Keystore** field.

      Note that the cacerts file may be "read only", in which case you must change its permissions so that it's writable.

5. Import the self-signed certificate generated above in this trust store:

   ```
   keytool -importcert -trustcacerts -alias alias -file certificate_file
   -keystore cacerts -storepass changeit
   ```

   Where:

   - *alias* is the WebCenter Portal alias (for example, webcenter_wls)
   - *certificate_file* is the file name for the certificate to export the key to (for example, webcenter_wls.cer)

   When prompted whether to trust the self-signed certificate, answer yes.

35.1.2 Configuring the Custom Identity and Java Trust Keystores

The next step is to configure the Custom Identity and Java Trust keystores on the WebCenter Portal server.

To configure the identity and trust keystores:

1. Log in to the WebLogic Server Administration Console.

   For information on logging into the WebLogic Server Administration Console, see Section 1.13.2, "Oracle WebLogic Server Administration Console."

2. Click the WebCenter Portal server (`WC_Spaces`) to configure the identity and trust keystores.

   The Settings pane for the WebCenter Portal server displays.

3. Open the **Configuration** tab, and then the **Keystores** subtab.

   The Keystores pane displays.

4. Click **Change**.

5. For **Keystores**, select `Custom Identity and Java Standard Trust` and click **Save**.

6. Under Identity, enter the path and filename of the **Custom Identity Keystore** you created in Section 35.1.1, "Creating the Custom Keystore."

7. Enter `JKS` as the **Custom Identity Keystore Type**.

8. Enter and confirm the **Custom Identity Keystore** password.

9. Under Trust, enter and confirm the **Java Standard Trust Keystore** password (typically set to `changeit`).

10. Click **Save** to save your entries.

11. Open the SSL tab.

12. Enter the **Private Key Alias** (for example, `webcenter_wls`).

13. Enter the **Private Key Passphrase** (for example, `MyPassword1`)

14. Click **Save** to save your entries.

35.1.3 Configuring the SSL Connection

To configure the SSL connection:

1. On the Settings pane for the WebCenter Portal server (`WC_Spaces`), open the Configuration tab and then the General subtab.

   The General Configuration pane displays.

2. Check **SSL Listen Port Enabled**.

3. Enter an **SSL Listen Port** number and click **Save**.

4. Open the SSL subtab and expand the Advanced options at the bottom of the page.

5. Check that the **Two Way Client Cert Behavior** option is set to `Client Certs Not Requested` and click **Save**.

6. Open the Control tab.

   The Control Settings pane displays.

7. Click **Restart SSL**.

8. Restart the WebLogic Server and open the SSL Portal URL.

   For a development or test environment only (that is, not for a production environment), if the hostname in the certificate does not match the host name, then the server must be started with:

   ```
   -Dweblogic.security.SSL.ignoreHostnameVerification=true
   ```

9. Accept the certificate for the session and log in.

35.2 Securing the Browser Connection to a Portal Framework Application with SSL

Securing the browser connection to a Portal Framework application uses the same configuration steps as for securing the browser connection to WebCenter Portal. The only difference is that the configuration occurs on the managed server that is hosting the Portal Framework application deployment rather than the `WC_Spaces` server. For more information, see Section 35.1, "Securing the Browser Connection to WebCenter Portal with SSL."

35.3 Securing the Connection from Oracle HTTP Server to WebCenter Portal with SSL

Securing the connection between the Oracle HTTP Server (OHS) and WebCenter Portal is described in the following sections:

- Section 35.3.1, "Configuring the Identity and Trust Keystores"
- Section 35.3.2, "Configuring the SSL Connection"
- Section 35.3.3, "Installing the Oracle HTTP Server"
- Section 35.3.4, "Wiring the WebCenter Portal Ports to the HTTP Server"
- Section 35.3.5, "Configuring the SSL Certificates"

35.3.1 Configuring the Identity and Trust Keystores

For instructions on how to configure the Identity and Trust keystores, see Section 35.1, "Securing the Browser Connection to WebCenter Portal with SSL."

35.3.2 Configuring the SSL Connection

To configure the SSL Connection:

1. On the Settings pane for the WebCenter Portal server, open the Configuration tab and then the General subtab.

   The General Configuration pane displays.

2. Check **SSL Listen Port Enabled**.
3. Enter an **SSL Listen Port** number and click **Save**.
4. On the **Configuration** tab, open the **SSL** subtab, and then expand the Advanced options at the bottom of the page.

   The SSL advanced options are displayed.

5. Set the **Two Way Client Cert Behavior** option to `Client Certs Not Requested` and click **Save**.
6. Open the Control tab on the Settings pane, and select the Start/Stop subtab.
7. Click **Restart SSL**.
8. Open the SSL WebCenter Portal URL.
9. Accept the certificate for the session and log in.
10. In the WLS Administration Console, click **View Changes and Restarts** on the Change Center pane and restart any affected servers or components.

35.3.3 Installing the Oracle HTTP Server

To install the Oracle HTTP Server:

1. Install the WebTier (see Section 33.2.3, "Installing and Configuring OAM").
   - Do not select WebCache; only select the HTTP Server.
   - Uncheck the checkbox to associate a WebLogic server during install.
2. Navigate to the `WT_ORACLE_HOME/instances/<your_instance>/bin` directory and start OHS using the following command:

   ```
   ./opmnctl startall
   ```

3. Check the status of OHS using the following command:

   ```
   ./opmnctl status -l
   ```

35.3.4 Wiring the WebCenter Portal Ports to the HTTP Server

To wire the WebCenter Portal ports to the HTTP server:

1. Open the file `WT_ORACLE_HOME/instances/<your_instance>/config/OHS/ohs1/mod_wl_ohs.conf`.
2. Add the following entry to `mod_wl_ohs.conf` to make WebCenter Portal work with OHS:

   ```
   <IfModule mod_weblogic.c>
           WebLogicHost host_id
           WebLogicPort port
           Debug OFF
           WLLogFile /tmp/ohs.log
           MatchExpression *.jsp
       </IfModule>

       <Location />
         SetHandler weblogic-handler
       </Location>
   ```

   Replacing *`host_id`* and *`port`* with the WebCenter Portal server ID and port number.
3. Open the file `WT_ORACLE_HOME/instances/<your_instance>/config/OHS/ohs1/ssl.conf`.
4. Add the following entry to `ssl.conf` to make WebCenter Portal run on the OHS SSL port:

   ```
   <Location />
           WebLogicHost host_id
           WebLogicPort port
           SetHandler weblogic-handler
           SecureProxy ON
           WLLogFile /tmp/ohs_ssl.log
           Debug ALL
           WlSSLWallet SSL_wallet
   </Location>

   <Location /webcenter>
         SetHandler weblogic-handler
   ```

```
        WebLogicHost host_id
        WebLogicPort port
      SecureProxy ON
        WLLogFile /tmp/ohs_ssl.log
        Debug ALL
        WlSSLWallet SSL_wallet
</Location>

<Location /webcenterhelp>
      SetHandler weblogic-handler
        WebLogicHost host_id
        WebLogicPort port
        SecureProxy ON
        WLLogFile /tmp/ohs_ssl.log
        Debug ALL
        WlSSLWallet SSL_wallet
</Location>

<Location /rsscrawl>
      SetHandler weblogic-handler
        WebLogicHost host_id
        WebLogicPort port
        SecureProxy ON
        WLLogFile /tmp/ohs_ssl.log
        Debug ALL
        WlSSLWallet SSL_wallet
</Location>

<Location /sesUserAuth>
      SetHandler weblogic-handler
        WebLogicHost host_id
        WebLogicPort port
        SecureProxy ON
        WLLogFile /tmp/ohs_ssl.log
        Debug ALL
        WlSSLWallet SSL_wallet
</Location>

<Location /rss>
      SetHandler weblogic-handler
        WebLogicHost host_id
        WebLogicPort port
        SecureProxy ON
        WLLogFile /tmp/ohs_ssl.log
        Debug ALL
        WlSSLWallet SSL_wallet
</Location>
```

Replacing *host_id* and *port* with the WebCenter Portal SSL server ID and port number (typically 8788), and *SSL_wallet* with the path to the WebLogic SSL wallet (for example, WT_ORACLE_HOME/instances/<your_instance>/config/OHS/ohs1/keystores/default).

Note: SSL should be configured at the server level rather than within the Location/directory sections. So, for example, instead of having:

```
<Location /mylocation>
  WLSSLWallet <walletfile>
  SecureProxy ON
</Location>
```

use:

```
SecureProxy ON
WlSSLWallet <walletfile>
```

at the server level (i.e., outside the Location/directory sections).

5. Go to `WT_ORACLE_HOME/instances/<your_instance>/bin` and start and check the status of OHS using the following commands:

```
./opmnctl stopall

./opmnctl startall
./opmnctl status -l
```

35.3.5 Configuring the SSL Certificates

To configure the SSL certificates:

1. For OHS to trust WebCenter Portal's certificate, the `WC_Spaces` certificate must be imported into the OHS trust store. Export the certificate from the `WC_Spaces` identity keystore:

```
keytool -exportcert -v -alias webcenter_wls -keystore webcenter_wls.jks
-storepass <password> -rfc -file webcenter_wls.cer
```

2. Navigate to `WT_ORACLE_HOME/instances/<your_instance>/config/OHS/ohs1/keystores/default` and run the following `orapki` command (typically located in `IDM_HOME`) to import the certificate into the wallet on the OHS side:

```
orapki wallet add -wallet . -trusted_cert -cert webcenter_wls.cer
-auto_login_only
```

   Note that `JAVA_HOME` should be set before running any orapki commands.

3. Determine the certificate `DN` by running the following command:

```
orapki wallet display -wallet wallet_location
```

4. For WebCenter Portal to trust OHS certificates, export the user certificate from OHS wallet and import it as a trusted certificate in the WebLogic trust store.

```
orapki wallet export -wallet . -cert cert.txt  -dn 'CN=\"Self-signed
Certificate for ohs1
\",OU=EXAMPLEORGUNIT,O=EXAMPLEORG,L=EXAMPLELOCATION,ST=CA,C=US'
```

5. Import the above certificate into the `WC_Spaces` managed server trust store available in `/scratch/wcwlsinstall/0408/wlshome/jrockit_160_05_R27.6.2-20/jre/lib/security/cacerts`:

```
keytool -file cert.txt -importcert -trustcacerts -alias ohs_cert
```

```
-keystore cacerts -storepass changeit
```

6. Restart OHS and the `WC_Spaces` server.

   You should now be able to access the SSL OHS, as well as the non-SSL OHS.

35.4 Securing the Browser Connection to the Discussions with SSL

Securing the browser connection to discussions with SSL is described in the following sections:

- Section 35.4.1, "Creating the Custom Keystore"
- Section 35.4.2, "Configuring the Identity and Trust Key Stores"
- Section 35.4.3, "Configuring the SSL Connection"

35.4.1 Creating the Custom Keystore

The first step is to generate a custom keystore as shown below:

1. Go to *`JDK_HOME`*`/bin/` and open a command prompt.
2. Using keytool, generate a key pair:

   ```
   keytool -genkeypair -keyalg RSA -dname "dname" -alias owc_discussions
   -keypass key_password -keystore owc_discussions.jks -storepass
   keystore_password -validity days_valid
   ```

   Where:

   - *`dname`* is the DN (distinguished name) to use (for example, `cn=customidentity,dc=owc_discussions,dc=example,dc=com`)
   - *`key_password`* is the password for the new public key, (for example, `MyPassword1`)
   - *`keystore_password`* is the keystore password, (for example, `MyPassword1`)
   - *`days_valid`* is the number of days for which the key password is valid (for example, `360`).

   Note: You must use the `-keyalg` parameter and specify `RSA` as its value as shown above as the default algorithm (DSA) used by `keytool` for generating the key is incompatible with Oracle WebServices Security Manager requirements.

3. Export the certificate containing the public key:

   ```
   keytool -exportcert -v -alias owc_discussions -keystore owc_discussions.jks
   -storepass keystore_password -rfc -file owc_discussions.cer
   ```

   Where:

   - *`keystore_password`* is the keystore password, (for example, `MyPassword1`)
4. Determine the trust store to use:

Since you are using a self-signed certificate, you must update it as a trusted certificate in the server trust store. To do this, you must determine your trust store by going to the server:

a. Log into the WebLogic Server Administration Console.

b. In the Domain Structure pane, expand Environments and click `Servers`.

c. In the list of servers, click `WC_Collaboration`.

d. Open the Configuration tab, and the Keystores subtab.

   The Keystores Settings pane displays.

e. Note down the location of the server in the **Java Standard Trust Keystore** field.

   Note that the `cacerts` file may be "read only," in which case you must change its permissions so that it's writable.

5. Import the self-signed certificate generated above in this trust store:

```
keytool -importcert -trustcacerts -alias owc_discussions
-file owc_discussions.cer -keystore cacerts -storepass changeit
```

   Note that the path to the `cacerts` file should be the absolute path. Otherwise, a new `cacerts` file will be created in the directory from where `keytool` is executed (to which the SSL port may not be able to listen).

   When prompted to trust the self-signed certificate, say `yes`.

35.4.2 Configuring the Identity and Trust Key Stores

To configure the identity and trust key stores:

1. Log in to the WebLogic Server Administration Console.

   For information on logging into the WebLogic Server Administration Console, see Section 1.13.2, "Oracle WebLogic Server Administration Console."

2. In the Domain Structure pane, expand **Environment** and click **Servers**.

   The Summary of Servers pane displays.

3. Click the Collaboration server (`WC_Collaboration`) to configure the identity and trust keystores.

   The Settings pane for the Collaboration server displays.

4. Open the **Configuration** tab, and then the **Keystores** subtab.

   The Keystores pane displays.

5. For **Keystores**, select **Custom Identity and Java Standard Trust**.

6. Under Identity, specify the keystore as `owc_discussions.jks`.

7. Set the keystore type to `JKS`.

8. Enter and confirm the keystore passphrase, (for example, `MyPassword1`)

9. Under Trust, set the **Java Standard Trust Keystore Passphrase** to `changeit` (this is fixed value) and click **Save**.

10. From the WLS Administration console, go to **Servers -> WC_Collaboration** and open the Configuration tab, and then the General subtab.

11. Check **SSL Port enabled**, specify a port that you want, and save your settings.

12. From the WLS Administration console, go to **Servers -> WC_Collaboration** and open the Configuration tab, and then the SSL subtab.
13. Specify the private key alias as `owc_discussions`, and set the password to MyPassword1.
14. Open the Control tab.

    The Control Settings pane displays.
15. Click **Restart SSL**.

35.4.3 Configuring the SSL Connection

To configure the SSL connection:

1. On the Settings pane for the Collaboration server, open the Configuration tab and then the General subtab.

   The General Configuration pane displays.
2. Check **SSL Listen Port Enabled**.
3. Enter an **SSL Listen Port** number and click **Save**.
4. On the **Configuration** tab, open the **SSL** subtab, and then expand the Advanced options at the bottom of the page.
5. Set the **Two Way Client Cert Behavior** option to `Client Certs Not Requested` and click **Save**.
6. Restart the `WC_Collaboration` server and open the SSL discussions URL at `https://host:port/owc_discussions`.
7. Accept the certificate for the session and log in.

35.5 Securing the WebCenter Portal Connection to Portlet Producers with SSL

Securing the connection to WSRP and PDK-Java portlet producers with SSL is described in the following sections:

- Section 35.5.1, "Creating the Custom Keystores"
- Section 35.5.2, "Configuring the Identity and Trust Key Stores"
- Section 35.5.3, "Configuring the SSL Connection"
- Section 35.5.4, "Registering the SSL-enabled WSRP Producer and Running the Portlets"
- Section 35.5.5, "Registering the SSL-enabled PDK-Java Producer and Running the Portlets"
- Section 35.5.6, "Consuming SSL-Enabled WSRP Portlets in JDeveloper"

35.5.1 Creating the Custom Keystores

For instructions on how to create the custom keystore, see Section 35.1.1, "Creating the Custom Keystore." Example commands for generating the keypair and exporting and importing the certificate are shown below:

```
./keytool -genkeypair -keyalg RSA -dname
"cn=customidentity,dc=portlet,dc=example,dc=com"
```

```
-alias portlet -keypass MyPassword1 -keystore portlet.jks -storepass MyPassword1
-validity 360

./keytool -exportcert -v -alias portlet -keystore portlet.jks -storepass
MyPassword1 -rfc -file portlet.cer

./keytool -importcert -trustcacerts -alias portlet -file portlet.cer
-keystore cacerts -storepass changeit
```

35.5.2 Configuring the Identity and Trust Key Stores

To configure the identity and trust key stores:

1. Log in to the WebLogic Server Administration Console.

   For information on logging into the WebLogic Server Administration Console, see Section 1.13.2, "Oracle WebLogic Server Administration Console."

2. In the Domain Structure pane, expand **Environment** and click **Servers**.

   The Summary of Servers pane displays.

3. Click the Portlet server (for example, `WC_Portlet`) to configure the identity and trust keystores.

   The Settings pane for the Portlet server displays.

4. Open the **Configuration** tab, and then the **Keystores** subtab.

   The Keystores pane displays.

5. For **Keystores**, select **Custom Identity and Java Standard Trust** and click **Save**.

6. Open the Control tab.

   The Control Settings pane displays.

7. Click **Restart SSL**.

35.5.3 Configuring the SSL Connection

To configure the SSL connection:

1. In the Domain Structure pane, expand **Environment** and select **Servers**.

2. Click the Portlet server (for example, `WC_Portlet`) for which you want to configure SSL.

3. Select **Configuration**.

4. Check **SSL Listen Port Enable**.

5. Enter a listen port number.

6. Select **Configuration** > **SSL**, and then open the Advanced options at the bottom of the page.

7. Select the **Two Way Client Cert Behavior** attribute and choose the **Client Certs Not Requested** option.

8. Click **Save**.

9. Restart the WebLogic Server and open the SSL URL.

10. Accept the certificate for the session and log in.

35.5.4 Registering the SSL-enabled WSRP Producer and Running the Portlets

To register the SSL-enabled WSRP producer and run the portlets:

1. Configure the `WC_Spaces` managed server to use the Custom Identity and Java Standard Trust store. This also uses the certificates in `JDK_HOME/jre/lib/security/cacerts`.
2. Download the certificate of the HTTPS producer URL and save it in `.PEM` format.

   Use Firefox 3.0 or later to download the certificate directly to `.PEM` format, or for other browsers use the WebLogic Server `der2pem` tool to convert to PEM format. For more information about using the `der2pem` tool, see the "der2pem" section in *Oracle Fusion Middleware Command Reference for Oracle WebLogic Server*. Note that WebLogic does not recognize any other format other than `.PEM` format.
3. Import the certificate into the `cacerts` file in `JDK_HOME/jre/lib/security` using the following keytool command:

   ```
   keytool -importcert -alias portlet_cert -file HOME/portlet_pem -keystore
   ./cacerts -storepass password
   ```

   Where:

   - *`portlet_cert`* is the portlet certificate alias
   - *`portlet_pem`* is the portlet certificate file (for example, `portlet_cert.pem`)
   - *`password`* is the keystore password
4. Restart `WC_Spaces`.
5. Start WLST as described in Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."
6. Connect to the Administration Server for the target domain with the following command:

   ```
   connect('user_name','password, 'host_id:port')
   ```

   Where:

   - *`user_name`* is the name of the user account with which to access the `WC_Spaces` server (for example, weblogic)
   - *`password`* is the password with which to access the `WC_Spaces` server
   - *`host_id`* is the host ID of the Administration server
   - *`port`* is the port number of the Administration server (for example, `7001`).
7. Run the `registerWSRPProducer` WLST command to register the producer:

   ```
   registerWSRPProducer('webcenter', 'sslwsrpprod','producer_wsdl')
   ```

   Where:

   - *`sslwsrpprod`* is the name of the SSL-enabled WSRP producer
   - *`producer_wsdl`* is the WSDL URL of the SSL-enabled WSRP producer

   For example:

   ```
   registerWSRPProducer('webcenter',
   'sslwsrpprod','https://example.com:7004/richtextportlet/portlets/wsrp2?WSDL')
   ```

8. Navigate to the HTTP or HTTPS WebCenter Portal URL.
9. Create a page and go to the Portlets link.
10. Go to the registered WSRP producer.
11. Add the portlet to the page.
12. Go to the view mode of the page and check that the WSRP portlet renders correctly.

35.5.5 Registering the SSL-enabled PDK-Java Producer and Running the Portlets

To register the SSL-enabled PDK-Java Producer and run the portlets:

1. Configure the WebCenter Portal managed server to use the Demo Identity and Trust store. This also uses the certificates in `JDK_HOME/jre/lib/security/cacerts`.
2. Log in to the WebLogic Server Administration Console.

   For information on logging into the WebLogic Server Administration Console, see Section 1.13.2, "Oracle WebLogic Server Administration Console."
3. On the Domain Structure pane, expand **Environment** and click **Servers**.

   The Summary of Servers pane displays.
4. Click `WC_Spaces` in the servers list.

   The Settings pane displays.
5. Open the Configuration tab and select the Keystores tab.
6. Make sure that the value for **Demo Identity and Demo Trust** is either `jks` or left blank.
7. Click **Save**.
8. Download the certificate of the HTTPS producer URL and save it in `.PEM` format.

   Use Firefox 3.0 or later to download the certificate directly to `.PEM` format, or for other browsers use the WebLogic Server `der2pem` tool to convert to PEM format. For more information about using the `der2pem` tool, see the "der2pem" section in *Oracle Fusion Middleware Command Reference for Oracle WebLogic Server*. Note that WebLogic does not recognize any other format other than `.PEM` format.
9. Import the certificate into the `cacerts` file in `JDK_HOME/jre/lib/security` using the following keytool command:

   ```
   keytool -importcert HOME/portlet_cert.pem -keystore ./cacerts -storepass
   changeit
   ```
10. Restart WC_Spaces.
11. Start WLST as described in Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."
12. Connect to the Administration Server for the target domain with the following command:

    ```
    connect('user_name','password, 'host_id:port')
    ```

    where:

    - *`user_name`* is the name of the user account with which to access the `WC_Spaces` server (for example, `weblogic`)

- *password* is the password with which to access the WC_Spaces server
- *host_id* is the host ID of the Administration server
- *port* is the port number of the Administration server (for example, 7001).

13. Run the registerPDKJavaProducer command:

```
registerPDKJavaProducer('webcenter', 'ssljpdkprod', 'producer_wsdl')
```

Where:

- *ssljpdkprod* is the name of the SSL-enabled PDK-Java producer
- *producer_wsdl* is the WSDL URL of the SSL-enabled PDK-Java producer

This enables one-way SSL for a Web producer. That is, only the server side (web producer) uses certificates. The Web producer code also uses a shared key feature (discussed later) for client authentication.

14. Go to the HTTP or HTTPS WebCenter Portal URL.
15. Create a page and go to the Portlets link.
16. Go to the registered PDK-Java producer.
17. Add the portlet to the page.
18. Go to the view mode of the page and check that the PDK-Java portlet renders correctly.

35.5.6 Consuming SSL-Enabled WSRP Portlets in JDeveloper

If you're consuming SSL-enabled portlets in JDeveloper, enable SSL on the producer's managed server as described in Section 35.5.3, "Configuring the SSL Connection."

1. For registration, the certificate (since it's a self-signed one) should be trusted by the JDeveloper runtime trust store. This means that you'll have to find out the JDK being used by the JDeveloper instance, and update the cacerts file (which is the trust store used by the JDeveloper instance) and then update the cacerts file using the following sample command:

```
<JDK_BIN>/keytool -importcert -trustcacerts -alias portlet_producer_cert -file
producer.cert -keystore ./cacerts -storepass changeit
```

2. The integrated WebLogic server typically uses a different trust store than JDeveloper. This trust store can be identified by accessing the console of the integrated WebLogic server:
   a. Access the WebLogic console.
   b. Expand the Environment node and then click the **Servers** node.
   c. Click **DefaultServer**.
   d. Open the Configurations tab, and then open the Keystores tab.

      The value for the Demo Trust Keystore attribute is the trust store used by the integrated WebLogic server.

   The trust store can also be located by looking at the integrated WebLogic server logs. To do this, start the integrated WebLogic server and search for "-Djavax.net.ssl.trustStore" in the logs to locate the trust store.

If you see this process parameter, then the integrated WebLogic server is using a trust store specified using JVM parameters, and you will need to import the certificate in this trust store using a command as per the following example:

```
<JDK6>/bin/keytool -importcert -trustcacerts -alias portlet_producer_cert
-file producer_cert.cert -keystore
<JDEV_MW_HOME>/wlserver_10.3/server/lib/DemoTrust.jks -storepass
DemoTrustKeyStorePassPhrase
```

35.6 Securing the WebCenter Portal Connection to the LDAP Identity Store

To configure the LDAP server port for SSL, refer to the appropriate administration documentation for the LDAP server. For Oracle Internet Directory (OID), an SSL port is installed by default. To use this port for LDAP communication from WebCenter Portal, the identity store should be configured for authentication with the appropriate authenticator. See Chapter 31, "Configuring the Identity Store" for the steps to do this for the identity store.

Note: When entering the Provider Specific information, be sure to specify an SSL port and to check the SSL Enabled checkbox.

If the `CA` is unknown to the Oracle WebLogic server, complete the two additional steps described in the following subsections:

- Section 35.6.1, "Exporting the OID Certificate Authority (CA)"
- Section 35.6.2, "Setting Up the WebLogic Server"

35.6.1 Exporting the OID Certificate Authority (CA)

If the `CA` is unknown to the Oracle WebLogic server (the command prompts the user to enter the keystore password) you must use `orapki` to create a certificate. The following example shows how to use this command to create the certificate `serverTrust.cert`:

```
orapki wallet export -wallet CA -dn "CN=myCA" -cert oid_server_trust.cert
```

35.6.2 Setting Up the WebLogic Server

If the `CA` is unknown to the Oracle WebLogic server, use the utility keytool to import the Oracle Internet Directory's CA into the WebLogic trust store. The following example shows how to use keytool to import the file `oid_server_trust.cert` into the server trust store `cacerts`:

```
keytool -importcert -v -trustcacerts -alias oid_server_trust -file
oid_server_trust.cer -keystore cacerts -storepass changeit
```

35.7 Securing the WebCenter Portal Connection to Content Server with SSL

If Content Server and the WebCenter Portal application in which you intend to create a repository connection are not on the same system or the same trusted private network, then identity propagation is not secure. To ensure secure identity propagation you must also configure SSL on Content Server.

Securing Content Server with SSL involves the following tasks:

- Section 35.7.1, "Configuring a Keystore and Key on the Client Side"
- Section 35.7.2, "Configuring a Keystore and Key on the Server Side"
- Section 35.7.3, "Verifying Signatures of Trusted Clients"
- Section 35.7.4, "Securing Identity Propagation"

In a production environment, Oracle recommends that you use only real certificates. For information about how to configure keystores when using real certificates, see the "Understanding Content Server Security Providers" chapter in *Oracle Fusion Middleware Administering Oracle WebCenter Content*.

35.7.1 Configuring a Keystore and Key on the Client Side

To configure a keystore on the WebCenter Portal application (client) side:

1. Go to the location, for example `jdk/bin`, where the keytool is located, and open the command prompt.
2. Generate the client keystore by running the following keytool command:

   ```
   keytool -genkey -keyalg RSA -validity 5000 -alias Client private key alias
   -keystore client-keystore.jks
   -dname "cn=client" -keypass Private key password -storepass KeyStore password
   ```

3. To verify that the keys have been correctly created, you can optionally run the following keytool command:

   ```
   keytool -list -keystore client-keystore.jks -storepass KeyStore password
   ```

4. To use the key, sign it by running the following keytool command:

   ```
   keytool -selfcert -validity 5000 -alias Client private key alias -keystore
   client-keystore.jks
   -keypass Private key password -storepass KeyStore password
   ```

5. Export the client public key by running the following keytool command:

   ```
   keytool -export -alias Client private key alias -keystore client-keystore.jks
   -file client.pubkey -keypass Private key password -storepass KeyStore password
   ```

35.7.2 Configuring a Keystore and Key on the Server Side

To configure a keystore on the Content Server side:

1. Go to the location, for example `jdk/bin`, where the keytool is located, and open the command prompt.
2. Generate the server keystore by running the following keytool command:

   ```
   keytool -genkey -keyalg RSA -validity 5000 -alias Server public key alias
   -keystore server-keystore.jks -dname "cn=server" -keypass Private server key
   password -storepass KeyStore password
   ```

3. To verify that the key has been correctly created, run the following keytool command:

   ```
   keytool -list -keystore server-keystore.jks -keypass Server private key
   password -storepass KeyStore password
   ```

4. To use the key, sign it by running the following keytool command:

```
keytool -selfcert -validity 5000 -alias Server public key alias -keystore
server-keystore.jks
-keypass Private server key password -storepass KeyStore password
```

5. Export the server public key to the server keystore by running the following keytool command:

```
keytool -export -alias Server public key alias -keystore server-keystore.jks
-file server.pubkey -keypass Server private key password -storepass KeyStore
password
```

35.7.3 Verifying Signatures of Trusted Clients

To verify signatures of trusted clients, import the client public key into the server keystore:

1. Go to the location, for example `jdk/bin`, where the keytool is located, and open the command prompt.
2. To verify the signature of trusted clients, import the client's public key in to the server keystore by running the following keytool command:

```
keytool -import -alias Client public key alias -file client.pubkey -keystore
server-keystore.jks -keypass Private server key password -storepass KeyStore
password
```

3. Import the server public key into the client keystore by running the following keytool command:

```
keytool -import -alias Server public key alias -file server.pubkey -keystore
client-keystore.jks -keypass Private key password -storepass KeyStore password
```

When the tool prompts you if the key is self-certified, you must enter `Yes`. Example 35–1 shows a sample output that is generated after this procedure is completed successfully.

Example 35–1 Sample Output Generated by the Keytool

```
[user@server]$ keytool -import -alias client -file client.pubkey
-keystore server-keystore.jks -keypass Server private key password -storepass
Keystore password
Owner: CN=client
Issuer: CN=client
Serial number: serial number, for example, 123a19cb
Valid from: Date, Year, and Time until: Date, Year, and Time
Certificate fingerprints:
        ...
Trust this certificate? [no]:  yes
Certificate was added to keystore.
```

35.7.4 Securing Identity Propagation

To secure identity propagation, you must configure SSL on Content Server.

1. Log on to Content Server as an administrator.
2. From **Administration**, select **Providers**.
3. On the Create a New Provider page, click **Add** for **sslincoming**.
4. On the Add Incoming Provider page, in **Provider Name**, enter a name for the provider, for example, `sslincomingprovider`.

When the new provider is set up, a directory with the provider name is created as a subdirectory of the *CONTENT_SERVER_HOME*/data/providers directory.

5. In **Provider Description**, briefly describe the provider, for example, `SSL Incoming Provider for securing the Content Server`.
6. In **Provider Class**, enter the class of the sslincoming provider, for example, `idc.provider.ssl.SSLSocketIncomingProvider`.

> **Note:** You can add a new SSL keepalive incoming socket provider or a new SSL incoming socket provider. Using a keepalive socket improves the performance of a session and is recommended for most implementations.

7. In **Connection Class**, enter the class of the connection, for example, `idc.provider.KeepaliveSocketIncomingConnection`.
8. In **Server Thread Class**, enter the class of the server thread, for example, `idc.server.KeepaliveIdcServerThread`.
9. In **Server Port**, enter an open server port, for example, `5555`.
10. Select the **Require Client Authentication** checkbox.
11. In **Keystore password**, enter the password to access the keystore.
12. In **Alias**, enter the alias of the keystore.
13. In **Alias password**, enter the password of the alias.
14. In **Truststore password**, enter the password of the trust store.
15. Click **Add**.

    The new incoming provider is now added.
16. Go to the new provider directory that was created in step 4.
17. To specify truststore and keystore, create a file named `sslconfig.hda`.
18. Copy the server keystore to the server.
19. Configure the `sslconfig.hda` file. Example 35–2 shows how the `.hda` file should look after you include the truststore and keystore information.

Example 35–2 Sample sslconfig.hda File

```
@Properties LocalData
TruststoreFile=/tmp/ssl/server_keystore
KeystoreFile=/tmp/ssl/server_keystore
@end
```

35.8 Securing the WebCenter Portal Connection to IMAP and SMTP with SSL

Before reconfiguring the mail server connection, you must first import the certificate into the trust store. Follow the steps below to put the certificate in the trust store and configure WebCenter Portal to use the trust store.

To secure the WebCenter Portal connection to IMAP and SMTP with SSL:

1. Open a browser and connect to your IMAP server with the following command:

```
https://imapserver:ssl_port
```

For example:

```
https:mailserver.example:993
```

2. Place your cursor on the page, right-click, and select **Properties**.
3. Click **Certificate**.
4. In the popup window, click the **Details** tab and click **Copy to File...**

   Be sure to use the `DER encoded binary(X.509)` format and copy to a file.
5. Convert the .DER format certificate to `.PEM` format.

   Use Firefox 3.0 or later to download the certificate directly to `.PEM` format, or for other browsers use the WebLogic Server `der2pem` tool to convert to PEM format. For more information about using the `der2pem` tool, see the "der2pem" section in *Oracle Fusion Middleware Command Reference for Oracle WebLogic Server*. Note that WebLogic does not recognize any other format other than `.PEM` format.
6. Import the certificate into the cacerts in the `JDK_HOME` using the following command:

   ```
   keytool -import -alias imap_cer -file cert_file.cer -keystore cacerts
   -storepass changeit
   ```

   Where *`cert_file`* is the name of the certificate file you downloaded.
7. Register the mail server connection as described in Section 15.4, "Registering Mail Servers."
8. Restart WebCenter Portal.
9. Log into WebCenter Portal and provide your mail credentials.

35.9 Securing a Portal Framework Application's Connection to IMAP and SMTP with SSL

To secure the connection to IMAP and SMTP with SSL for a Portal Framework application:

1. Follow the steps in Section 35.8, "Securing the WebCenter Portal Connection to IMAP and SMTP with SSL" up to and including step 7.
2. Add the following property to the truststore:

   ```
   -Djavax.net.ssl.trustStore=C:\jive\mailtool\jssecacerts
   -Djavax.net.ssl.trustStorePassword=changeit
   ```

   For example:

   ```
   set JAVA_PROPERTIES=-Dplatform.home=%WL_HOME% -Dwls.home=%WLS_HOME%
   -Dweblogic.home=%WLS_HOME%
   -Djavax.net.ssl.trustStore=C:\jive\mailtool\jssecacerts
   -Djavax.net.ssl.trustStorePassword=changeit
   ```
3. Restart the Portal Framework application.
4. Log into the application and provide your mail credentials.

35.10 Securing the Connection to Oracle SES with SSL

There are two scenarios in which you may want to configure SSL for SES: The first scenario is where WebCenter Portal or a Portal Framework application has already been protected with SSL but SES has not; the second scenario is where SES has been protected with SSL, but WebCenter Portal or the Portal Framework application has not. These two scenarios are described in the following subsections:

- Section 35.10.1, "Securing Oracle SES with SSL"
- Section 35.10.2, "Securing the Connection to Oracle SES with SSL"

35.10.1 Securing Oracle SES with SSL

In this scenario, WebCenter Portal or your Portal Framework application is already protected with SSL, but SES is not. Follow the steps below to secure SES with SSL.

Before registering the SES connection, you must first import the certificate into the Trust Store. Follow the steps below to put the certificate in the Trust Store and register the Oracle Secure Enterprise Search (SES) connection.

To download the certificate of the HTTPS URL and save it:

1. Configure SSL on the WebCenter side using the following certificate name:

   ```
   cn=<myhost>
   ```

   where `<myhost>` is the fully qualified name of the host where WebCenter is installed.

   For more information about configuring SSL on WebCenter Portal, see Section 35.1, "Securing the Browser Connection to WebCenter Portal with SSL." For more information about configuring SSL for a Portal Framework application, see Section 35.2, "Securing the Browser Connection to a Portal Framework Application with SSL."

2. Export the WebCenter certificate in PEM format (i.e., `<myhost>.crt`).

   You can use Firefox 3.0 or later to download the certificate directly to `.PEM` format. For other browsers, follow the steps below and then use the WebLogic Server `der2pem` tool to convert to PEM format.

   a. Click **Certificate**.

   b. In the popup window, open the Details tab, and click **Copy to File...**

      Use **DER encoded binary(X.509)** format and copy the certificate to a file.

   c. Convert the .DER format certificate to .PEM format.

      For more information about using the `der2pem` tool, see the "der2pem" section in *Oracle Fusion Middleware Command Reference for Oracle WebLogic Server*. Note that WebLogic does not recognize any other format other than `.PEM` format.

3. In SES, import the certificate into the following keystores:

   - `<SES Installation Directory>/jdk6/jre/lib/security/cacerts`
   - `<SES Installation Directory>/seshome/jdk/jre/lib/security/cacerts`

   using the following command:

   ```
   keytool -importcert -trustcacerts -alias webcenter_wls -file <myhost>.crt
   ```

```
-keystore cacerts -storepass changeit
```

4. In SES, create a source for Oracle WebCenter in which the crawl and authorization endpoints point to the WebCenter Portal or Portal Framework application's HTTPS ports.
5. Create a schedule and source group for the crawl (see Section 18.5.1.2, "Configuring Search Parameters and Crawlers Using Fusion Middleware Control").
6. Finish the WebCenter-side configuration for SES and restart SES and WebCenter Portal or your Portal Framework application.
7. Create some objects in WebCenter Portal or your Portal Framework application and start the crawl.
8. After the crawl has been completed, search for a keyword and the results should appear in WebCenter.

35.10.2 Securing the Connection to Oracle SES with SSL

In this scenario, WebCenter Portal or your Portal Framework application is not protected with SSL, but SES is.

To download the certificate of the HTTPS URL and save it:

1. Use your browser to navigate to the Web Services URL that Oracle Secure Enterprise Search exposes to enable search requests at:

   `http://host:port/search/query/OracleSearch`

   For example:

   `https://example.com:7777/search/query/OracleSearch`

2. Place your cursor on the page, right-click with your mouse, and select **Properties**.
3. Click **Certificate**.
4. In the popup window, open the Details tab, and click **Copy to File...**

   Use **DER encoded binary(X.509)** format and copy the certificate to a file.

5. Convert the .DER format certificate to .PEM format.

   Use Firefox 3.0 or later to download the certificate directly to `.PEM` format, or for other browsers use the WebLogic Server `der2pem` tool to convert to PEM format. For more information about using the `der2pem` tool, see the "der2pem" section in *Oracle Fusion Middleware Command Reference for Oracle WebLogic Server*. Note that WebLogic does not recognize any other format other than `.PEM` format.

6. Import the certificate into `DemoTrustKeyStore.jks` or cacerts in the `JDK_HOME` using the following command:

   ```
   keytool -import -alias ses_cer -file cert_file.cer -keystore cacerts -storepass
   changeit
   ```

   where *`cert_file`* is the name of the certificate file you downloaded.

7. Register the SES connection as described in Section 18.4.2, "Registering Oracle Secure Enterprise Search Servers."
8. Restart WebCenter Portal or your Portal Framework application.

35.11 Securing the WebCenter Portal Connection to Microsoft Live Communication Server and Office Communication Server with SSL

To secure the WebCenter Portal connection to Microsoft Live Communication Server (LCS) or Office Communication Server 2007 (OCS) with SSL, follow the steps below to import the certificate into the trust store, and point WebCenter Portal to use the trust store. Note that securing the WebCenter Portal connection to Microsoft Live Communication Server or Office Communication Server with SSL is optional since they can be configured with confidentiality using WS-Security.

Before registering the LCS or OCS connection, you must first import the certificate into the trust store. Follow the steps below to put the certificate in the trust store:

1. Open your browser and go to the communication server (for example, `https://example.com/RTC`)
2. Place your cursor on the page, right-click, and select **Properties**.
3. Click **Certificate**.
4. In the popup window, open the **Details** tab and click **Copy to File...**

   Use Firefox 3.0 or later to download the certificate directly to `.PEM` format, or for other browsers use the WebLogic Server `der2pem` tool to convert to PEM format. For more information about using the `der2pem` tool, see the "der2pem" section in *Oracle Fusion Middleware Command Reference for Oracle WebLogic Server*. Note that WebLogic does not recognize any other format other than `.PEM` format.
5. Import the certificate into the `cacerts` using the following keytool command:

   ```
   keytool -import -alias lcs_cer -file cert_file.cer -keystore cacerts -storepass changeit
   ```

   where *`cert_file`* is the name of the certificate file you downloaded.
6. Locate the `cacerts` file used by the communication server in the installation, and also update the communication server referenced `cacerts` file with this certificate:

   ```
   keytool -import -alias lcs_cer -file cert_file.cer -keystore cacerts -storepass changeit
   ```
7. Register the communication server connection as described in Section 14.3, "Registering Instant Messaging and Presence Servers."
8. Restart the WebCenter Portal server.

35.12 Securing the WebCenter Portal Connection to an External BPEL Server with SSL

This section describes how to secure the WebCenter Portal connection to a BPEL server when the BPEL server resides in an external SOA domain.

> **Note:** When SOA is installed in an external domain, the Identity Asserter and Authenticator should be configured exactly as for WebCenter Portal. For more information on configuring the Identity Asserter and Authenticator for an external LDAP identity store, see Section 31.1, "Reassociating the Identity Store with an External LDAP Server."

To secure the WebCenter Portal connection to an external BPEL server with SSL:

1. Copy the public certificate (`webcenter_wls.cer`) from WebCenter Portal into the SOA domain.
2. Go to *`JDK_HOME`*`/bin/` and open a command prompt.
3. Generate a custom keystore on the SOA domain naming the keystore `soa_server1.jks`, and the alias `soa_server1` using the following `keytool` command:

   ```
   keytool -genkeypair -keyalg RSA -dname dname -alias soa_soa_server1 -keypass
   key_pass -keystore soa_server1.jks -storepass keystore_password -validity
   days_valid
   ```

   Where:

   - *`dname`* is the DN (distinguished name) to use (for example, `cn=customidentity,dc=example,dc=com`)
   - *`key_pass`* is the password for the new public key, (for example, `MyPassword1`)
   - *`keystore_password`* is the keystore password, (for example, `MyPassword1`)
   - *`days_valid`* is the number of days for which the key password is valid (for example, `360`).
4. Export the certificate from `soa_wls.jks` using the following command:

   ```
   keytool -exportcert -v -alias soa_server1 -keystore soa_server1.jks
   -storepass keystore_password -rfc -file soa_server1.cer
   ```

   Where:

   - *`keystore_password`* is the keystore password, (for example, `MyPassword1`)
5. Log in to the WebLogic Server Administration Console on the SOA domain.

   For information on logging into the WebLogic Server Administration Console, see Section 1.13.2, "Oracle WebLogic Server Administration Console."
6. In the Navigation pane, expand **Environment** and click **Servers**.

   The Summary of Servers pane displays.
7. From the Configuration tab, click `soa_server1` in the list of servers.

   The Settings page for `soa_server1` displays.
8. Open the Keystores tab.

   The Keystore settings for `soa_server1` displays.
9. For **Keystores**, select `Custom Identity and Java Standard Trust`.
10. Specify the path and filename of keystore (`soa_server1.jks`) created above.
11. Go to the directory containing the java standard trust (cacerts file) specified in the **Java Standard Trust Keystores** field and import the SOA and WebCenter Portal public certificates into this file so they may be trusted by the server:

    ```
    keytool -importcert -trustcacerts -alias webcenter_wls -file webcenter_wls.cer
    -keystore cacerts -storepass keystore_password

    keytool -importcert -trustcacerts -alias soa_server1 -file soa_server1.cer
    ```

```
-keystore cacerts -storepass keystore_password
```

Where:

- *keystore_password* is the keystore password, (for example, MyPassword1)

Say yes when prompted to trust the certificate.

12. From the WLS Administration Console on the SOA domain, open the SSL tab.

    The SSL settings for soa_server1 display.

13. Specify soa_server1 as the **Private Key Alias**.

14. Enter and confirm the password for the private key (for example, MyPassword1) and click **Save**.

15. Open the General tab.

    The General settings for soa_server1 display.

16. Make sure that **Listen Port Enabled** is not selected.

17. Select **SSL Listen Port Enabled**, specify the **SSL Listen Port**, and click **Save**.

18. Open the Control tab, and then open the Start/Stop sub-tab.

    The Start/Stop settings for soa_server1 display.

19. Select soa_server1 from the list of servers, and click **Restart SSL**.

20. Restart the soa_server1 Managed Server on the SOA domain.

21. From the WebCenter Portal domain, import the soa_server1.cer certificate as a trusted certificate to the server trust store (cacerts) using the following keytool commands:

    ```
    keytool -importcert -trustcacerts -alias soa_server1 -file soa_server1.cer
    -keystore cacerts -storepass changeit
    ```

    Say yes when prompted to trust the certificate.

22. Add the Worklist connection on the WebCenter Portal domain as described in Section 20.4.2, "Registering Worklist Connections" specifying the host:ssl_port settings for soa_server1 when defining the BPEL URL.

23. Restart the WC_Spaces server.

36

Configuring WS-Security

This chapter describes how to set up WS-Security for WebCenter Portal and Portal Framework applications and related services and components based on your topology. This section covers the following configurations:

- A simple topology, with the WebCenter Portal or Portal Framework applications and all components sharing the same domain
- A typical topology, with the WebCenter Portal or Portal Framework applications and components divided across two domains
- A complex topology, with the WebCenter Portal or Portal Framework applications and components divided across multiple domains

Within these three topologies, configuration is described for the WebCenter Portal or Portal Framework applications, discussions, worklists, and WSRP producers. These configurations and the steps for securing applications consuming the WebCenter Portal API are covered in the following sections:

- Section 36.1, "Configuring WS-Security for a Simple Topology"
- Section 36.2, "Configuring WS-Security for a Typical Topology"
- Section 36.3, "Configuring WS-Security for a Complex Topology"
- Section 36.4, "Securing WebCenter Portal for Applications Consuming WebCenter Portal Client API with WS-Security"

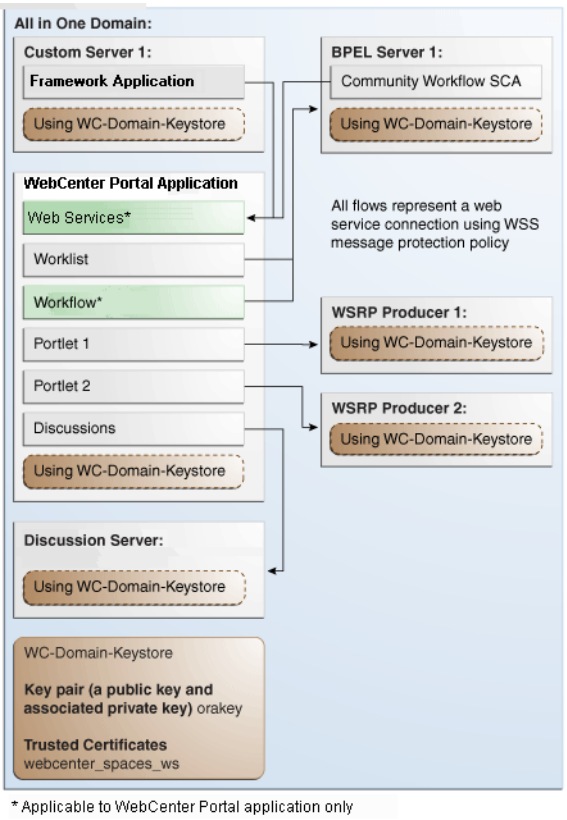
Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console. Users with the `Monitor` or `Operator` roles can view security information but cannot make changes.

See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

36.1 Configuring WS-Security for a Simple Topology

This section describes how to configure WS-Security for a topology where the WebCenter Portal and Portal Framework applications, the BPEL server, and WSRP producers share the same domain (Figure 36–1).

Figure 36–1 WS-Security for a Simple Configuration



The steps to configure WS-Security for a simple single-domain topology are described in the following sections:

- Section 36.1.1, "Roadmap to Configuring WS-Security for a Simple Topology"
- Section 36.1.2, "Setting Up the WebCenter Portal Domain Keystore"
- Section 36.1.3, "Configuring the Discussions Server for a Simple Topology"
- Section 36.1.4, "Command Summary for a Simple Topology"

36.1.1 Roadmap to Configuring WS-Security for a Simple Topology

The flow chart (Figure 36–1) and table (Table 36–1) in this section provide an overview of the prerequisites and tasks required to configure WS-Security for a simple single-domain topology.

Figure 36–2 Configuring WS-Security for a Simple Topology

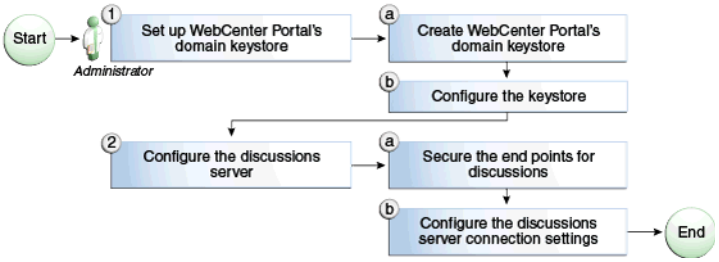


Table 36–1 shows the tasks and sub-tasks to configure WS-Security for a simple topology.

Table 36–1 Configuring WS-Security for a Simple Topology

Actor	Task	Sub-task	Notes
Administrator	**1.** Set up the WebCenter Portal domain keystore	**1.a** Create the WebCenter Portal domain keystore	
		1.b Configure the keystore	
	2. Configure the discussions server	**2.a** Secure the discussions end points	
		2.b Configure the discussions server connection settings	

36.1.2 Setting Up the WebCenter Portal Domain Keystore

The security credentials of the WebCenter Portal application, discussions server, BPEL server, and WSRP producers can be retrieved and managed using a Java Keystore (JKS). A keystore is a file that provides information about available public and private keys. Keys are used for a variety of purposes, including authentication and data integrity. User certificates and the trust points needed to validate the certificates of peers are also stored securely in the wallet or keystore. See *Oracle Fusion Middleware Security and Administrator's Guide for Web Services* for information about JKS.

This section contains the following subsections:

- Section 36.1.2.1, "Creating the WebCenter Portal Domain Keystore"
- Section 36.1.2.2, "Configuring the Keystore with WLST"
- Section 36.1.2.3, "Configuring the Keystore Using Fusion Middleware Control"

36.1.2.1 Creating the WebCenter Portal Domain Keystore

This section describes how to create a keystore and keys using a Java Keystore (JKS). JKS is the proprietary keystore format defined by Sun Microsystems. To create and manage the keys and certificates in the JKS, use the `keytool` utility that is distributed with the Java JDK 6.

To create the WebCenter Portal domain keystore:

1. Go to *`JDK_HOME`*`/jdk/bin` and open a command prompt.
2. Using keytool, generate a key pair:

```
keytool -genkeypair -keyalg RSA -dname "consumer_dname" -alias orakey
-keypass key_password -keystore keystore -storepass keystore_password -validity
days_valid
```

Where:

- *`consumer_dname`* is the name of the consumer. This can be any string as long as it's in the correct format (for example, `cn=spaces,dc=example,dc=com`)
- *`key_password`* is the password for the new public key, (for example, `MyPassword`)
- *`keystore`* is the keystore name, (for example, `default-keystore.jks`)
- *`keystore_password`* is the keystore password, (for example, `MyPassword`)
- *`days_valid`* is the number of days for which the key password is valid (for example, `1064`).

Example 36–1 Generating the Keypair

```
keytool -genkeypair -keyalg RSA -dname "cn=spaces,dc=example,dc=com" -alias orakey
-keypass MyPassword -keystore default-keystore.jks -storepass MyPassword -validity
1064
```

> **Note:** You must use the `-keyalg` parameter and specify `RSA` as its value as shown above as the default algorithm (DSA) used by `keytool` for generating the key is incompatible with Oracle Web Services Security Manager requirements.

3. Export the certificate containing the public key:

   ```
   keytool -exportcert -v -alias orakey -keystore keystore -storepass
   keystore_password -rfc -file orakey.cer
   ```

   Where:

   - *`keystore`* is the keystore name, (for example, `default-keystore.jks`)
   - *`keystore_password`* is the keystore password, (for example, `MyPassword`)

Example 36–2 Exporting the Certificate Containing the Public Key

```
keytool -exportcert -v -alias orakey -keystore default-keystore.jks -storepass
MyPassword -rfc -file orakey.cer
```

4. Import the certificate with the alias `webcenter_spaces_ws` (choose **Yes** when prompted whether to overwrite the existing certificate with the alias `orakey`):

   ```
   keytool -importcert -alias webcenter_spaces_ws -file orakey.cer
   -keystore default-keystore.jks -storepass keystore_password
   ```

   Where:

   - *`keystore_password`* is the keystore password

Example 36–3 Importing the Certificate

```
keytool -importcert -alias webcenter_spaces_ws -file orakey.cer -keystore
default-keystore.jks -storepass MyPassword
```

5. Continue by configuring the keystore using either WLST as described in Section 36.1.2.2, "Configuring the Keystore with WLST," or using Fusion Middleware Control as described in Section 36.1.2.3, "Configuring the Keystore Using Fusion Middleware Control."

   Table 36–2 shows the keystore contents you should wind up with after creating and configuring the keystore.

Table 36–2 Portal Domain Keystore Contents for a Simple Topology

Key Alias	Description
`orakey`	Key pair used to sign and encrypt outbound messages from WebCenter Portal. This key is used by both OWSM (portlets and worklist) and discussions.

Table 36–2 (Cont.) Portal Domain Keystore Contents for a Simple Topology

Key Alias	Description
webcenter_spaces_ws	Certificate containing the public key for the orakey private key used in the WebCenter Portal domain. The certificate is used to encrypt outbound Web service messages from the workflow application on BPEL Server1 in the WebCenter Portal domain, to the Web service API on WebCenter Portal domain.

36.1.2.2 Configuring the Keystore with WLST

After creating the keystore, configure the keystore service and update the credential store so that OWSM can read the keystore and keys correctly. You can do this using either Fusion Middleware control, as described in Section 36.1.2.3, "Configuring the Keystore Using Fusion Middleware Control," or using WLST as described below.

To configure the credential store:

1. Go to the *<DOMAIN_HOME>*/config/fmwconfig directory, and open the file jps-config.xml in an editor.
2. Locate the <serviceInstance node for the keystore.provider Provider:

```
<serviceInstance name="keystore" provider="keystore.provider"
location="./default-keystore.jks">
<description>Default JPS Keystore Service</description>
```

3. Make sure that the default-keystore.jks keystore file is copied to the *<DOMAIN_HOME>*/config/fmwconfig directory, and that the location is specified as ./default-keystore.jks:

```
<serviceInstance name="keystore" provider="keystore.provider"
location="./default-keystore.jks">
<description>Default JPS Keystore Service</description>
```

4. Use the following WLST commands to update the credential store:

```
updateCred(map="oracle.wsm.security", key="keystore-csf-key", user="owsm",
password=keystore_password, desc="Keystore key")
updateCred(map="oracle.wsm.security", key="enc-csf-key", user="orakey",
password=private_key_password, desc="Encryption key")
updateCred(map="oracle.wsm.security", key="sign-csf-key", user="orakey",
password=private_key_password, desc="Signing key")
```

Where:

- *keystore_password* is the keystore password specified in step 2 of Section 36.1.2.1, "Creating the WebCenter Portal Domain Keystore," (for example, MyPassword)
- *private_key_password* is the private key password specified in step 2 of Section 36.1.2.1, "Creating the WebCenter Portal Domain Keystore," (for example, MyPassword)

Example 36–4 Updating the Credential Store

```
updateCred(map="oracle.wsm.security", key="keystore-csf-key", user="owsm",
password="MyPassword", desc="Keystore key")
updateCred(map="oracle.wsm.security", key="enc-csf-key", user="orakey",
password="MyPassword", desc="Encryption key")
updateCred(map="oracle.wsm.security", key="sign-csf-key", user="orakey",
password="MyPassword", desc="Signing key")
```

5. Restart all servers.

36.1.2.3 Configuring the Keystore Using Fusion Middleware Control

After creating the keystore, configure the keystore service and update the credential store so that OWSM can read the keystore and keys correctly. You can do this using either WLST, as described in Section 36.1.2.2, "Configuring the Keystore with WLST," or using Fusion Middleware control as described below.

To configure the keystore provider:

1. Ensure that the `default-keystore.jks` keystore file is copied to the *<DOMAIN_HOME>*`/config/fmwconfig` directory, and then specify the location as `./default-keystore.jks`.
2. Open Fusion Middleware Control and log in to the WebCenter Portal domain.

   For information on logging in to Fusion Middleware Control, see Chapter 6, "Starting Enterprise Manager Fusion Middleware Control."
3. In the Navigation pane, expand the WebLogic Domain node and click the WebCenter Portal domain (`wc_domain` by default).
4. From the WebLogic Domain menu, select **Security -> Security Provider Configuration**.

   The Security Provider Configuration page displays.
5. Expand the Keystore section on the Security Provider Configuration page.
6. Click **Configure**.

   The Keystore Configuration page displays.
7. Use the following settings to specify the location of the keystore that contains the certificate and private key, and the signature key and encryption key aliases:
   - **Keystore Path**: `./default-keystore.jks`
   - **Password**: Enter and confirm the password for the keystore.
   - **Key Alias**: `orakey`
   - **Signature Password**: Enter and confirm the password for the signature key.
   - **Crypt Alias**: `orakey`
   - **Crypt Password**: Enter and confirm the password for the encryption key.
8. Click **OK** to save your settings.
9. Restart the Administration server for the domain.

36.1.3 Configuring the Discussions Server for a Simple Topology

In a simple topology, the discussions server is in the same domain as the `WC_Spaces` server and consequently no extra keystore configuration is needed since the keystore configured for the WebCenter Portal domain is used for the discussions as well. However, for production environments you should protect the discussions Web service endpoints with an OWSM policy and configure the discussions server connection settings. These configuration steps are described in the following subsections:

- Section 36.1.3.1, "Attaching Security Policies for WebCenter Portal and Discussions Web Service Endpoints"

- Section 36.1.3.2, "Securing the Discussions End Points"
- Section 36.1.3.3, "Configuring the Discussions Server Connection Settings"

> **Note:** Discussions-specific Web services messages sent by WebCenter Portal and Portal Framework applications to the discussions server are not encrypted. For message confidentiality, the discussions server URL must be accessed over Secure Socket Layer (SSL). For more information, see Chapter 35, "Configuring SSL."

36.1.3.1 Attaching Security Policies for WebCenter Portal and Discussions Web Service Endpoints

In a new or patched WebCenter Portal instance, the assigned security policy configuration is set to "no security policy." You must attach Oracle Web Services Manager (OWSM) security policies for the WebCenter Portal Web service endpoint and the discussions authenticated Web service endpoint. For a production environment, continue by hardening the security by following the steps in Section 36.1.3.2, "Securing the Discussions End Points."

> **Note:** In a patched WebCenter Portal instance, you must determine the policy names before you patch, then attach the policies after you patch. For required steps, see the "Patching Oracle WebCenter Portal" section in *Oracle Fusion Middleware Patching Guide*.

To attach the Web service security policy configuration in a new instance:

> **Note:** For clustered environments, repeat these steps for each of the managed servers where WebCenter Portal and discussions are deployed.

1. Ensure that the `WC_Spaces` and `WC_Collaboration` managed servers are running.
2. Run the following WLST command to attach an OWSM policy on the WebCenter Portal Web service endpoint:

```
attachWebServicePolicy(application='webcenter', moduleName='webcenter',
moduleType='web', serviceName='SpacesWebService',
subjectName='SpacesWebServiceSoapHttpPort',
policyURI='oracle/wss11_saml_token_with_message_protection_service_policy')
```

3. Run the following WLST command to attach an OWSM policy on the discussions Web service endpoint:

```
attachWebServicePolicy(application='owc_discussions',
moduleName='owc_discussions', moduleType='web',
serviceName='OWCDiscussionsServiceAuthenticated',
subjectName='OWCDiscussionsServiceAuthenticated',
policyURI='oracle/wss10_saml_token_service_policy')
```

4. Restart the `WC_Spaces` and `WC_Collaboration` managed servers.

36.1.3.2 Securing the Discussions End Points

The discussions Web service endpoints require user identity to be propagated for calls originating from WebCenter Portal. For a production environment, the Web service endpoints must be secured with OWSM policies to ensure that messages are not tampered with, and can't be viewed by others while in transit. To do this, both the public access Web service endpoint and authenticated user access endpoint should be secured with the appropriate OWSM policies using either Fusion Middleware Control or WLST.

This section contains the following subsections:

- Section 36.1.3.2.1, "Securing the Discussions Server End Points Using Fusion Middleware Control"
- Section 36.1.3.2.2, "Securing the Discussions Server End Points Using WLST"

36.1.3.2.1 Securing the Discussions Server End Points Using Fusion Middleware Control

To secure the discussions end points using Fusion Middleware Control, follow the steps below:

1. Log in to Fusion Middleware Control and from the Navigation pane, expand **WebCenter> Portal> Discussions** and click `Discussions(WC_Collaboration)`.

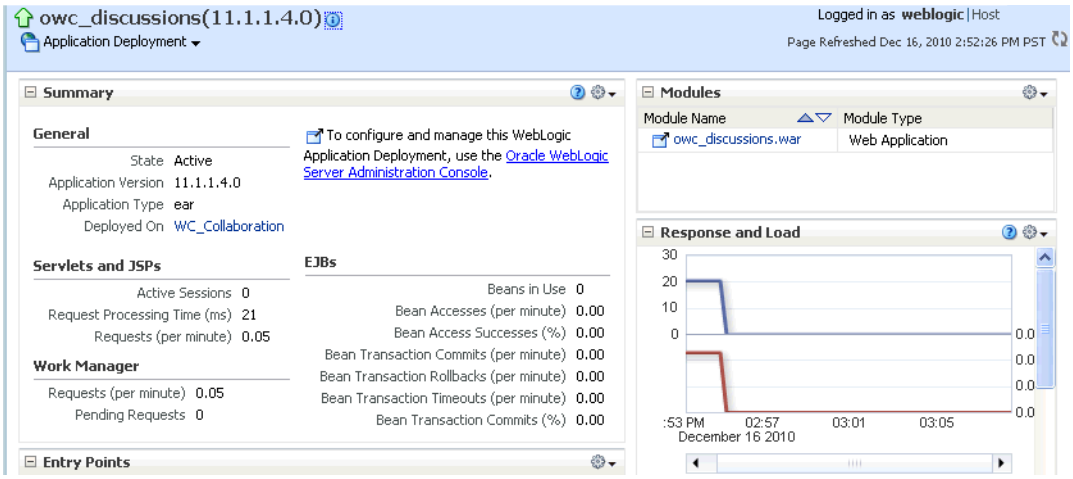
   The discussions home page displays (see Figure 36–3).

Figure 36–3 Discussions Home Page

2. Click the `owc_discussions` target.

   The home page for the `owc_discussions` application displays (see Figure 36–4).

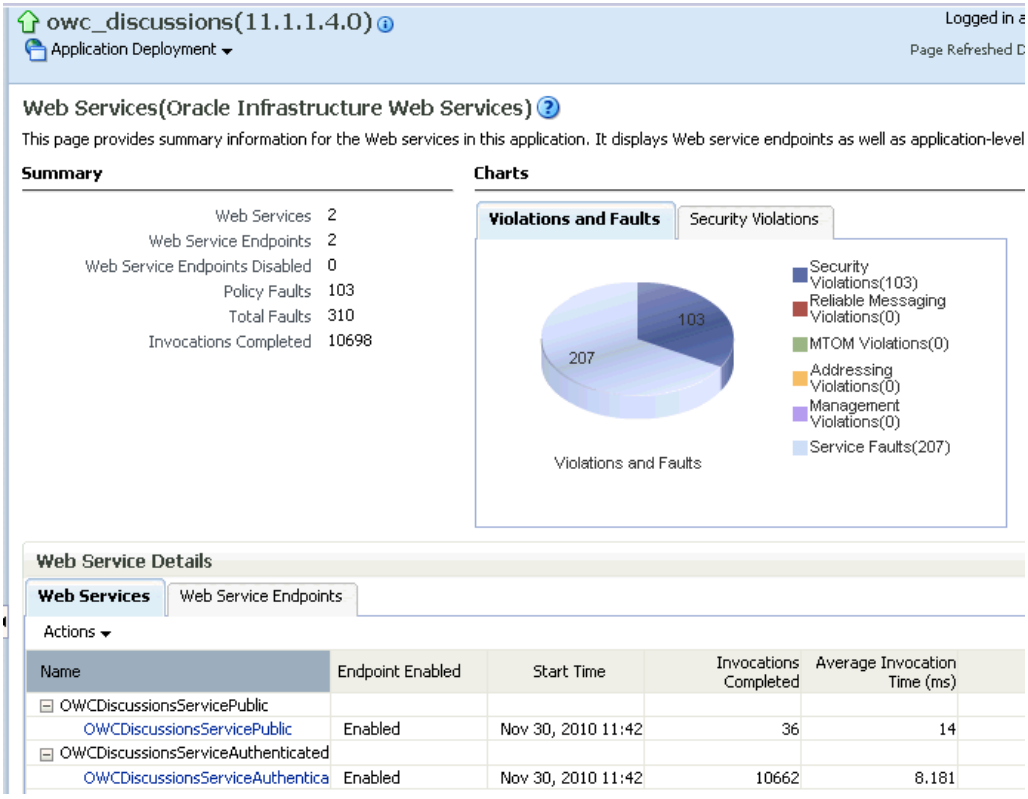
Figure 36–4 owc_discussions Home Page



3. From the Application Deployment menu, select **Web Services**.

   The Web Services page for the `owc_discussions` application displays (see Figure 36–5).

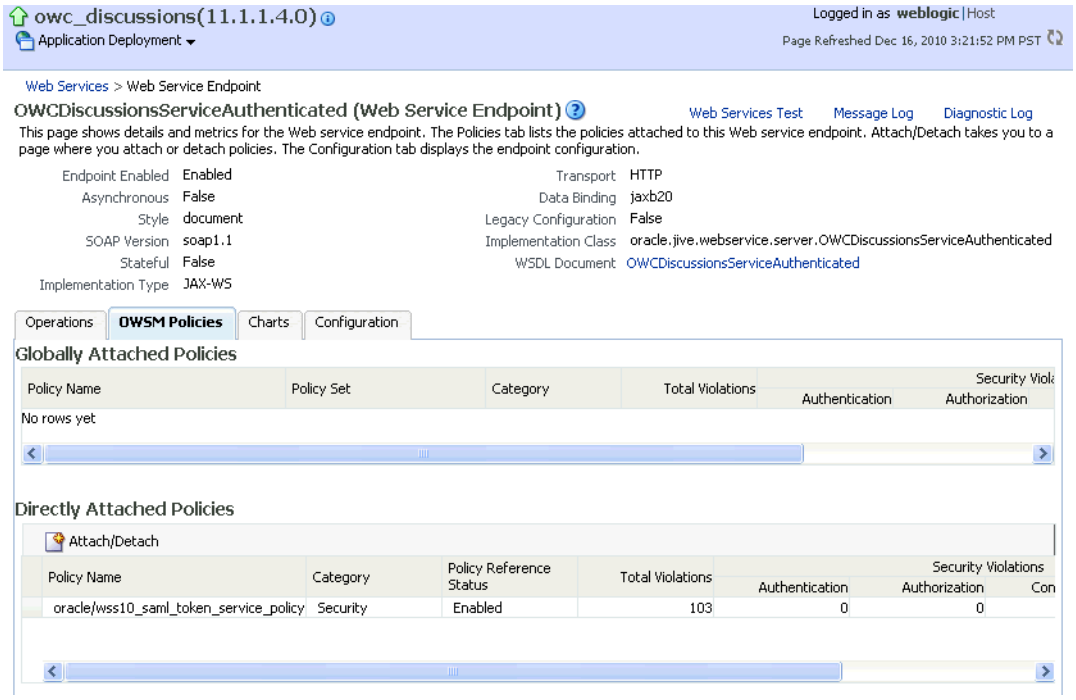
Figure 36–5 Web Services Page for owc_discussions



4. Open the Web Services tab, and click the `OWCDiscussionsServiceAuthenticated` Web Service end point.

   The Web Service Endpoint page for owc_discussions displays (see Figure 36–6).

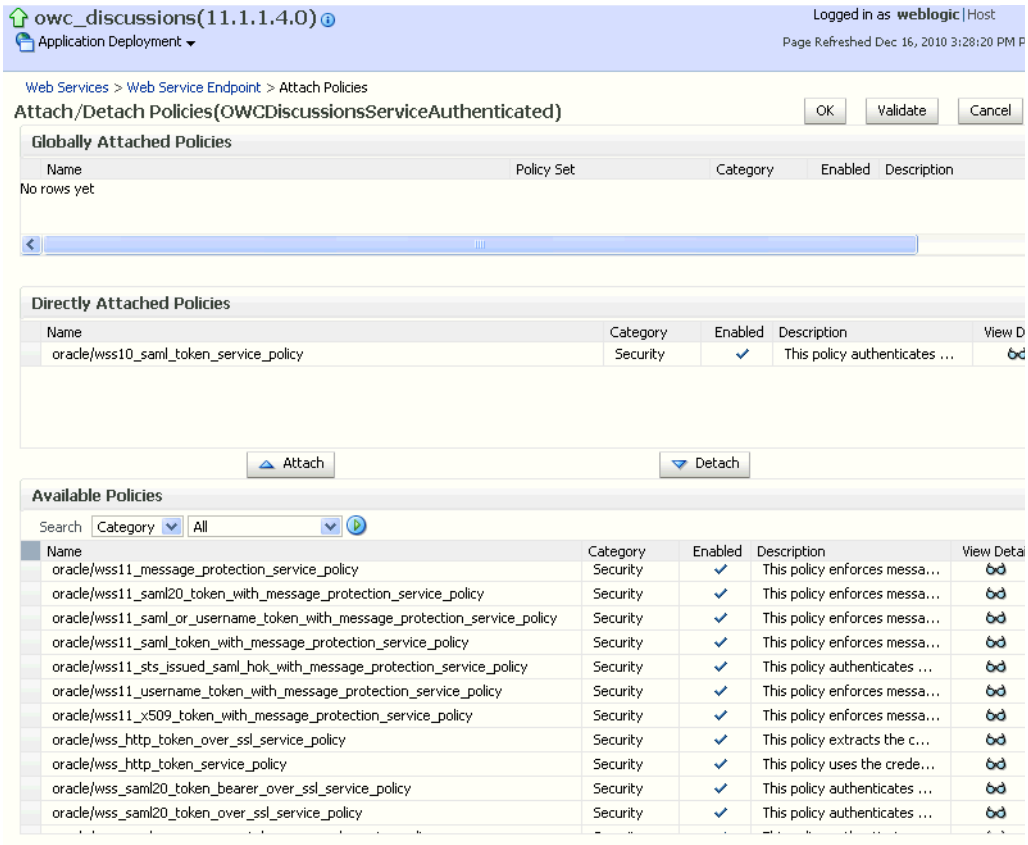
Figure 36–6 Web Service Endpoint Page



5. Click **Attach/Detach**.

   The Attach Policy page displays (see Figure 36–7).

Figure 36–7 Attach Policy Page



6. Use the **Attach** and **Detach** buttons to attach `oracle/wss11_saml_token_with_message_protection_service_policy` and detach `oracle/wss10_saml_token_service_policy`.
7. Click **OK**.

36.1.3.2.2 Securing the Discussions Server End Points Using WLST

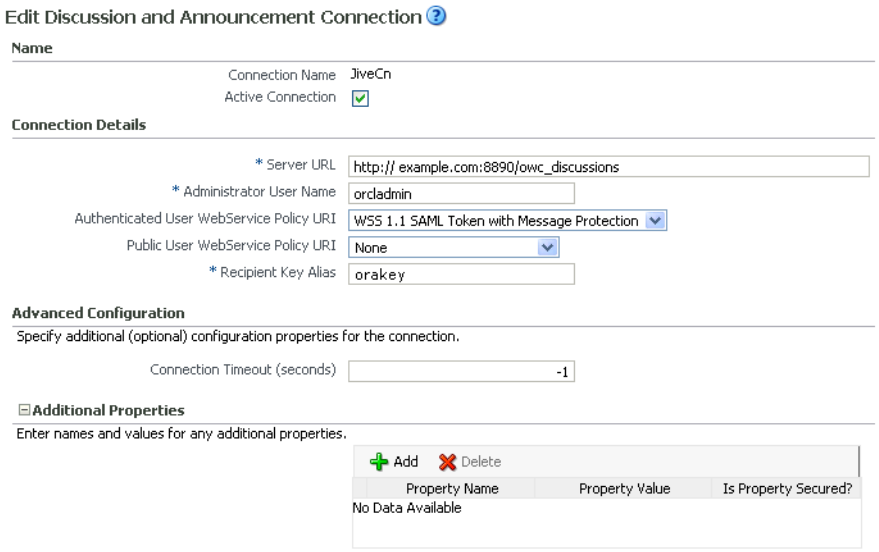
To secure the discussions server endpoints using WLST, detach the `wss10_saml_token_service_policy` and attach the `wss11_saml_token_with_message_protection_service_policy` using the following WLST commands:

```
detachWebServicePolicy(application='owc_discussions',
moduleName='owc_discussions', moduleType='web',
serviceName='OWCDiscussionsServiceAuthenticated',
subjectName='OWCDiscussionsServiceAuthenticated',
policyURI='oracle/wss10_saml_token_service_policy')


attachWebServicePolicy(application='owc_discussions',
moduleName='owc_discussions', moduleType='web',
serviceName='OWCDiscussionsServiceAuthenticated',
subjectName='OWCDiscussionsServiceAuthenticated',
policyURI='oracle/wss11_saml_token_with_message_protection_service_policy')
```

36.1.3.3 Configuring the Discussions Server Connection Settings

You must supply the WS-Security client certificate information within the discussions server connection that is configured for your WebCenter Portal or Portal Framework application, as described in Section 12.3, "Registering Discussions Servers." Figure 36–8 shows example connection detail settings for the Edit Discussions and Announcement Connection page.

Figure 36–8 Edit Discussions and Announcement Connection Page



36.1.4 Command Summary for a Simple Topology

Use the following command summary to quickly configure the keystore for a simple topology.

Generate the Keystore

Use the following `keytool` commands to generate the keystore, replacing the values in bold with those for your local environment:

```
keytool -genkeypair -keyalg RSA -dname "cn=spaces,dc=example,dc=com" -alias orakey
-keypass MyPassword -keystore default-keystore.jks -storepass MyPassword -validity
1064

keytool -exportcert -v -alias orakey -keystore default-keystore.jks -storepass
MyPassword -rfc -file orakey.cer

keytool -importcert -alias webcenter_spaces_ws -file orakey.cer
-keystore default-keystore.jks -storepass MyPassword
```

When prompted that the certificate already exists, say `yes`.

```
keytool -importcert -alias df_orakey_public -file orakey.cer
-keystore owc_discussions.jks -storepass MyPassword
```

Copy the `default-keystore.jks` file to your `domain_home/config/fmwconfig` directory.

Configure the Keystore

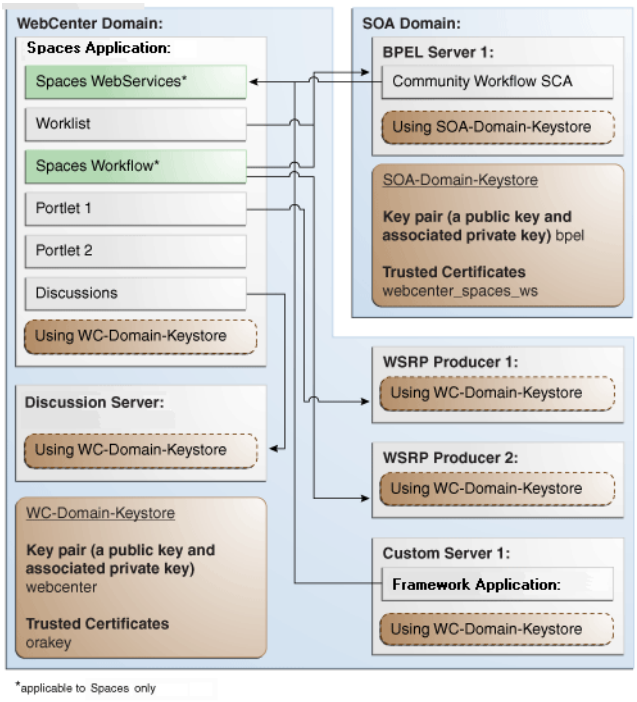
Using WLST, connect to the WebCenter Portal domain as an administrator and run the following commands:

```
updateCred(map="oracle.wsm.security", key="keystore-csf-key", user="owsm",
password="MyPassword", desc="Keystore key")
updateCred(map="oracle.wsm.security", key="enc-csf-key", user="orakey",
password="MyPassword", desc="Encryption key")
updateCred(map="oracle.wsm.security", key="sign-csf-key", user="orakey",
password="MyPassword", desc="Signing key")
```

36.2 Configuring WS-Security for a Typical Topology

This section describes how to configure WS-Security for a topology where the WebCenter Portal application and the WSRP producers share the same domain, but the BPEL server is in an external domain - the SOA domain (see Figure 36–9).

Figure 36–9 WS-Security for a Typical Configuration



The steps to configure WS-Security for a typical two domain topology are described in the following sections:

- Section 36.2.1, "Roadmap to Configuring WS-Security for a Typical Topology"
- Section 36.2.2, "Setting Up the WebCenter Portal Domain Keystore"
- Section 36.2.3, "Configuring the Discussions Server for a Typical Topology"
- Section 36.2.4, "Setting Up the SOA Domain"
- Section 36.2.5, "Command Summary for a Typical Topology"

36.2.1 Roadmap to Configuring WS-Security for a Typical Topology

The flow chart (Figure 36–10) and table (Table 36–3) in this section provide an overview of the prerequisites and tasks required to configure WS-Security for a typical two domain topology.

Figure 36–10 Configuring WS-Security for a Typical Topology

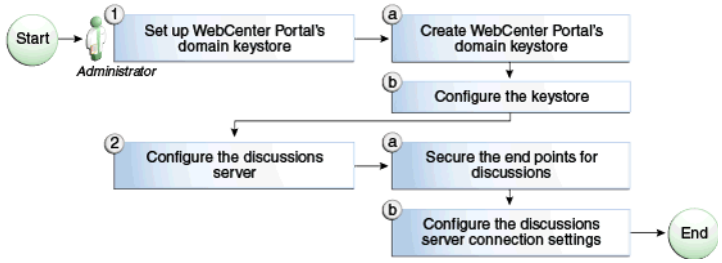


Table 36–3 shows the tasks and sub-tasks to configure WS-Security for a typical two domain topology.

Table 36–3 Configuring WS-Security for a Typical Topology

Actor	Task	Sub-task	Notes
Administrator	**1.** Set up the WebCenter Portal domain keystore	**1.a** Create the WebCenter Portal domain keystore	
		1.b Configure the keystore	
	2. Configure the discussions server	**2.a** Secure the discussions end points	
		2.b Configure the discussions server connection settings	

36.2.2 Setting Up the WebCenter Portal Domain Keystore

The security credentials of a WebCenter Portal application, discussions server, BPEL server (in a separate domain), and WSRP producers can be retrieved and managed using a Java Keystore (JKS). A keystore is a file that provides information about available public and private keys. Keys are used for a variety of purposes, including authentication and data integrity. User certificates and the trust points needed to validate the certificates of peers are also stored securely in the wallet or keystore. See *Oracle Fusion Middleware Security and Administrator's Guide for Web Services* for information about JKS.

This section contains the following subsections:

- Section 36.2.2.1, "Creating the WebCenter Portal Domain Keystore"
- Section 36.2.2.2, "Configuring the Keystore Using WLST"
- Section 36.2.2.3, "Configuring the Keystore Using Fusion Middleware Control"

36.2.2.1 Creating the WebCenter Portal Domain Keystore

This section describes how to create a keystore and keys using a Java Keystore (JKS). JKS is the proprietary keystore format defined by Sun Microsystems. To create and manage the keys and certificates in the JKS, use the `keytool` utility that is distributed with the Java JDK 6.

To create the WebCenter Portal domain keystore:

1. Go to *`JDK_HOME`*`/jdk/bin` and open a command prompt.

2. Using keytool, generate a key pair:

```
keytool -genkeypair -keyalg RSA -dname "consumer_dname" -alias webcenter
-keypass key_password -keystore keystore -storepass keystore_password -validity
days_valid
```

Where:

- *consumer_dname* is the name of the consumer (for example, `cn=spaces,dc=example,dc=com`)
- *key_password* is the password for the new public key, (for example, `MyPassword`)
- *keystore* is the keystore name, (for example, `webcenter.jks`)
- *keystore_password* is the keystore password, (for example, `MyPassword`)
- *days_valid* is the number of days for which the key password is valid (for example, `1064`).

Example 36–5 Generating the Keypair

```
keytool -genkeypair -keyalg RSA -dname "cn=spaces,dc=example,dc=com" -alias
webcenter -keypass MyPassword -keystore webcenter.jks -storepass MyPassword
-validity 1064
```

> **Note:** You must use the `-keyalg` parameter and specify `RSA` as its value as shown above as the default algorithm (DSA) used by `keytool` for generating the key is incompatible with Oracle Web Services Security Manager requirements.

3. Export the certificate containing the public key:

```
keytool -exportcert -v -alias webcenter -keystore keystore
-storepass keystore_password -rfc -file webcenter_public.cer
```

Where:

- *keystore* is the keystore name, (for example, `webcenter.jks`)
- *keystore_password* is the keystore password, (for example, `MyPassword`)

Example 36–6 Exporting the Certificate Containing the Public Key

```
keytool -exportcert -v -alias webcenter -keystore webcenter.jks
-storepass MyPassword -rfc -file webcenter_public.cer
```

4. Continue by configuring the keystore using either WLST, as described in Section 36.2.2.2, "Configuring the Keystore Using WLST," or Fusion Middleware Control, as described in Section 36.2.2.3, "Configuring the Keystore Using Fusion Middleware Control."

Table 36–4 shows the keystore contents you should wind up with after creating and configuring the keystore.

Table 36–4 WebCenter Portal Domain Keystore Contents for a Typical Topology

Key Alias	Description
`webcenter`	Key pair used to sign and encrypt outbound messages from WebCenter Portal. This key is used by both OWSM (portlets and worklist) and discussions.
`orakey`	Certificate containing the public key for the `BPEL` private key used in the SOA domain. The certificate is used to encrypt outbound Web service messages from the workflow application on BPEL Server1 in the WebCenter Portal domain, to the worklist component to the SOA server on the SOA domain.

36.2.2.2 Configuring the Keystore Using WLST

After creating the WebCenter Portal domain keystore, configure the keystore service and update the credential store so that OWSM can read the keystore and keys correctly. You can do this either using Fusion Middleware Control, as described in Section 36.2.2.3, "Configuring the Keystore Using Fusion Middleware Control," or using WLST as described below.

To configure the keystore service:

1. Go to the *<DOMAIN_HOME>*`/config/fmwconfig` directory, and open the file `jps-config.xml` in an editor.
2. Locate the `<serviceInstance` node for the keystore.provider `Provider`
3. Ensure that the `webcenter.jks` keystore file is copied to the *<DOMAIN_HOME>*`/config/fmwconfig` directory, and then specify the location as `./webcenter.jks`.
4. Use the following WLST commands to update the credential store:

```
updateCred(map="oracle.wsm.security", key="keystore-csf-key", user="owsm",
password=keystore_password, desc="Keystore key")
updateCred(map="oracle.wsm.security", key="enc-csf-key", user="webcenter",
password=private_key_password, desc="Encryption key")
updateCred(map="oracle.wsm.security", key="sign-csf-key", user="webcenter",
password=private_key_password, desc="Signing key")
```

   Where:

   - *keystore_password* is the keystore password specified in step 2 of Section 36.2.2.1, "Creating the WebCenter Portal Domain Keystore," (for example, `MyPassword`)
   - *private_key_password* is the private key password specified in step 2 of Section 36.2.2.1, "Creating the WebCenter Portal Domain Keystore," (for example, `MyPassword`)

Example 36–7 Updating the Credential Store

```
updateCred(map="oracle.wsm.security", key="keystore-csf-key", user="owsm",
password="MyPassword", desc="Keystore key")
updateCred(map="oracle.wsm.security", key="enc-csf-key", user="webcenter",
password="MyPassword", desc="Encryption key")
updateCred(map="oracle.wsm.security", key="sign-csf-key", user="webcenter",
password="MyPassword", desc="Signing key")
```

5. Restart all servers.

36.2.2.3 Configuring the Keystore Using Fusion Middleware Control

After creating the keystore, configure the keystore service and update the credential store so that OWSM can read the keystore and keys correctly. You can do this either using WLST, as described in Section 36.2.2.2, "Configuring the Keystore Using WLST," or using Fusion Middleware Control as described below.

To configure the keystore provider:

1. Open Fusion Middleware Control and log in to the WebCenter Portal domain.

   For information on logging in to Fusion Middleware Control, see Chapter 6, "Starting Enterprise Manager Fusion Middleware Control."

2. In the Navigation pane, expand the WebLogic Domain node and click the WebCenter Portal domain (`wc_domain` by default).

3. From the WebLogic Domain menu, select **Security -> Security Provider Configuration**.

   The Security Provider Configuration page displays.

4. Expand the Keystore section on the Security Provider Configuration page.

5. Click **Configure**.

   The Keystore Configuration page displays.

6. Use the following settings to specify the location of the keystore that contains the certificate and private key, and the signature key and encryption key aliases:

   - **Keystore Path**: `./webcenter.jks`
   - **Password**: Enter and confirm the password for the keystore.
   - **Key Alias**: `webcenter`
   - **Signature Password**: Enter and confirm the password for the signature key.
   - **Crypt Alias**: `webcenter`
   - **Crypt Password**: Enter and confirm the password for the encryption key.

7. Click **OK** to save your settings.

8. Restart the Administration server for the domain.

36.2.3 Configuring the Discussions Server for a Typical Topology

Configuring the discussions server for a typical topology is exactly the same as for a simple topology. For more information, see Section 36.1.3, "Configuring the Discussions Server for a Simple Topology."

36.2.4 Setting Up the SOA Domain

This section describes how to set up the SOA domain keystore and contains the following subsections:

- Section 36.2.4.1, "Creating the SOA Domain Keystore"
- Section 36.2.4.2, "Configuring the Keystore Using WLST"
- Section 36.2.4.3, "Configuring the Keystore Using Fusion Middleware Control"

36.2.4.1 Creating the SOA Domain Keystore

This section describes how to create a SOA domain keystore and keys using a Java Keystore (JKS).

To create the SOA domain keystore:

1. Go to *JDK_HOME*/jdk/bin and open a command prompt.
2. Create a keystore by importing the public certificate (webcenter_public.cer) from the WebCenter Portal domain:

   ```
   keytool -importcert -alias webcenter_spaces_ws -file webcenter_public.cer
    -keystore bpel.jks -storepass keystore_password
   ```

   Where:

   - *keystore_password* is the keystore password, (for example, MyPassword)

Example 36–8 Importing the Public Certificate

```
keytool -importcert -alias webcenter_spaces_ws -file webcenter_public.cer
-keystore bpel.jks -storepass MyPassword
```

3. Using keytool, create a keypair to be used in the SOA domain for signing and encrypting messages:

   ```
   keytool -genkeypair -keyalg RSA -dname "consumer_dname" -alias bpel
   -keypass key_password -keystore keystore -storepass keystore_password
   -validity days_valid
   ```

   Where:

   - *consumer_dname* is the name of the consumer (for example, cn=bpel,dc=example,dc=com)
   - *key_password* is the password for the new public key, (for example, MyPassword)
   - *keystore* is the keystore name, (for example, bpel.jks)
   - *keystore_password* is the keystore password, (for example, MyPassword)
   - *days_valid* is the number of days for which the key password is valid (for example, 1064).

Example 36–9 Generating the Keypair

```
keytool -genkeypair -keyalg RSA -dname "cn=bpel,dc=example,dc=com" -alias bpel
-keypass MyPassword -keystore bpel.jks -storepass MyPassword -validity 1064
```

> **Note:** You must use the -keyalg parameter and specify RSA as its value as shown above as the default algorithm (DSA) used by keytool for generating the key is incompatible with Oracle Web Services Security Manager requirements.

4. Export the certificate so it can be imported in the WebCenter Portal domain using the orakey alias:

   ```
   keytool -exportcert -v -alias bpel -keystore keystore -storepass
   keystore_password -rfc -file orakey.cer
   ```

   Where:

- *keystore* is the keystore name, (for example, `webcenter.jks`)
- *keystore_password* is the keystore password, (for example, `MyPassword`)

Example 36–10 Exporting the Certificate Containing the Public Key

```
keytool -exportcert -v -alias bpel -keystore bpel.jks -storepass MyPassword -rfc
-file orakay.cer
```

5. Import the certificate with a different alias (choose **Yes** when prompted whether to overwrite the existing certificate with the alias `orakey`):

   ```
   keytool -importcert -alias orakey -file orakey.cer -keystore webcenter.jks
   -storepass keystore_password
   ```

   Where:

   - *keystore_password* is the keystore password

Example 36–11 Importing the Certificate

```
keytool -importcert -alias orakey -file orakay.cer -keystore webcenter.jks
-storepass MyPassword
```

36.2.4.2 Configuring the Keystore Using WLST

After creating the SOA domain keystore, configure the keystore service and update the credential store so that OWSM can read the keystore and keys correctly. You can do this either with Fusion Middleware Control, as described in Section 36.2.4.3, "Configuring the Keystore Using Fusion Middleware Control," or using WLST as described below.

To configure the keystore service:

1. Go to the *<DOMAIN_HOME>*`/config/fmwconfig` directory, and open the file `jps-config.xml` in an editor.
2. Locate the `<serviceInstance` node for the keystore.provider `Provider`
3. Ensure that the `bpel.jks` keystore file is copied to the *<DOMAIN_HOME>*`/config/fmwconfig` directory, and then specify the location as `./bpel.jks`.
4. Use the following WLST commands to configure the credential store:

   ```
   updateCred(map="oracle.wsm.security", key="keystore-csf-key", user="owsm",
   password="MyPassword", desc="Keystore key")
   updateCred(map="oracle.wsm.security", key="enc-csf-key", user="bpel",
   password="MyPassword", desc="Encryption key")
   updateCred(map="oracle.wsm.security", key="sign-csf-key", user="bpel",
   password="MyPassword", desc="Signing key")
   ```

5. Restart all servers.

36.2.4.3 Configuring the Keystore Using Fusion Middleware Control

After creating the keystore, configure the keystore service and update the credential store so that OWSM can read the keystore and keys correctly. You can do this either with WLST, as described in Section 36.2.4.2, "Configuring the Keystore Using WLST," or using Fusion Middleware Control as described below.

To configure the keystore provider:

1. Open Fusion Middleware Control and log in to the SOA domain.

For information on logging in to Fusion Middleware Control, see Chapter 6, "Starting Enterprise Manager Fusion Middleware Control."

2. In the Navigation pane, expand the WebLogic Domain node and click the SOA domain.
3. From the WebLogic Domain menu, select **Security -> Security Provider Configuration**.
4. Expand the Keystore section on the Security Provider Configuration page.
5. Click **Configure**.

   The Keystore Configuration page displays.
6. Use the following settings to specify the location of the keystore that contains the certificate and private key, and the signature key and encryption key aliases:
   - **Keystore Path**: `./bpel.jks`
   - **Password**: Enter and confirm the password for the keystore.
   - **Key Alias**: `bpel`
   - **Signature Password**: Enter and confirm the password for the signature key.
   - **Crypt Alias**: `bpel`
   - **Crypt Password**: Enter and confirm the password for the encryption key.
7. Click **OK** to save your settings.
8. Restart the Administration server for the domain.

36.2.5 Command Summary for a Typical Topology

Use the following command summary to quickly configure the keystore for a typical topology.

Generate the Keystore

Use the following `keytool` commands to generate the keystore, replacing the values in bold with those for your local environment:

```
keytool -genkeypair -keyalg RSA -dname "cn=spaces,dc=example,dc=com" -alias
webcenter -keypass MyPassword -keystore webcenter.jks -storepass MyPassword
-validity 1064

keytool -exportcert -v -alias webcenter -keystore webcenter.jks
-storepass MyPassword -rfc -file webcenter_public.cer

keytool -importcert -alias webcenter_spaces_ws -file webcenter_public.cer
-keystore bpel.jks -storepass MyPassword
```

When prompted that the certificate already exists, say `yes`.

```
keytool -genkeypair -keyalg RSA -dname "cn=bpel,dc=example,dc=com" -alias bpel
-keypass MyPassword -keystore bpel.jks -storepass MyPassword -validity 1024

keytool -exportcert -v -alias bpel -keystore bpel.jks -storepass MyPassword
-rfc -file orakay.cer

keytool -importcert -alias orakey -file orakay.cer -keystore webcenter.jks
-storepass MyPassword
```

When prompted to trust the certificate, say `yes`.

```
keytool -importcert -alias df_webcenter_public -file webcenter_public.cer
-keystore owc_discussions.jks -storepass MyPassword
```

When prompted to trust the certificate, say `yes`.

Copy the `webcenter.jks` file to your `domain_home/config/fmwconfig` directory, and the `bpel.jks` file to your `soa_domain_home/config/fmwconfig` directory.

Configure the WebCenter Portal Domain Keystore

Follow the steps below to configure the service instance reference for the WebCenter Portal domain:

1. Navigate to the `<DOMAIN_HOME>/config/fmwconfig` directory.
2. Copy `webcenter.jks` to the `<DOMAIN_HOME>/config/fmwconfig` directory if you haven't already done so.
3. Open `jps-config.xml` in an editor.
4. Locate `<serviceInstance node for keystore.provider Provider`.
5. Specify the location as `./webcenter.jks`.
6. Using WLST, connect to the WebCenter Portal domain as an admin user and run the following commands:

```
updateCred(map="oracle.wsm.security", key="keystore-csf-key", user="owsm",
password="MyPassword", desc="Keystore key")
updateCred(map="oracle.wsm.security", key="enc-csf-key", user="webcenter",
password="MyPassword", desc="Encryption key")
updateCred(map="oracle.wsm.security", key="sign-csf-key", user="webcenter",
password="MyPassword", desc="Signing key")
```

Configure the SOA Domain Keystore

Follow the steps below to configure service instance reference for the SOA domain:

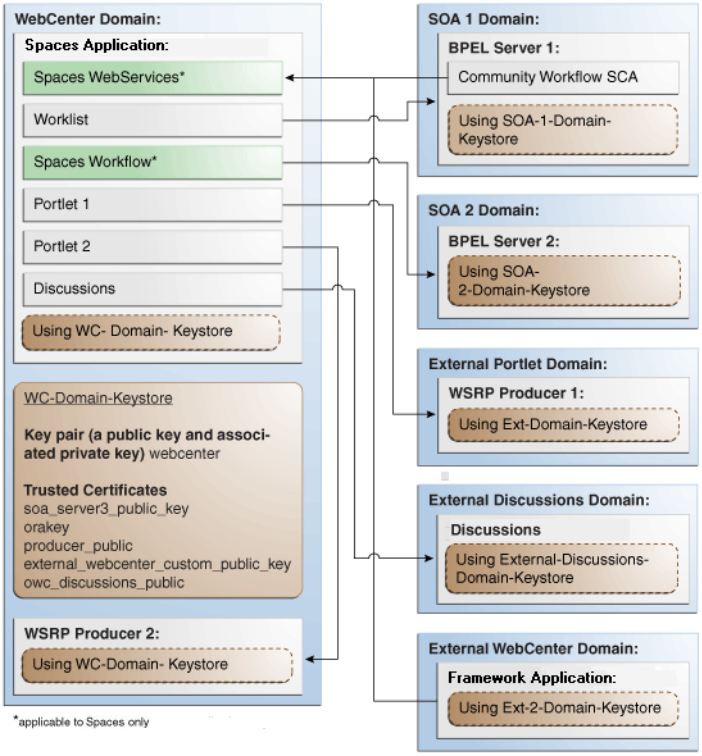
1. Navigate to the `<SOA_DOMAIN_HOME>/config/fmwconfig` directory.
2. Copy `bpel.jks` to the `<SOA_DOMAIN_HOME>/config/fmwconfig` directory if you haven't done already done so.
3. Open `jps-config.xml` in an editor.
4. Locate `<serviceInstance node for keystore.provider Provider`.
5. Specify the location as `./bpel.jks`.
6. Using WLST, connect to the SOA domain as an admin user and run the following commands:

```
updateCred(map="oracle.wsm.security", key="keystore-csf-key", user="owsm",
password="MyPassword", desc="Keystore key")
updateCred(map="oracle.wsm.security", key="enc-csf-key", user="bpel",
password="MyPassword", desc="Encryption key")
updateCred(map="oracle.wsm.security", key="sign-csf-key", user="bpel",
password="MyPassword", desc="Signing key")
```

36.3 Configuring WS-Security for a Complex Topology

This section describes how to configure WS-Security for a complex topology where the WebCenter Portal application, the discussions server, and a WSRP producer are in the same domain, two BPEL servers are in separate SOA domains, and one WSRP producer is in an external portlet domain (see Figure 36–11).

Figure 36–11 WS-Security for a Complex Configuration



The steps to configure WS-Security for a complex topology with multiple domains are described in the following sections:

- Section 36.3.1, "Roadmap to Configuring WS-Security for a Complex Topology"
- Section 36.3.2, "Setting Up the WebCenter Portal Domain Keystores"
- Section 36.3.3, "Configuring the Discussions Server for a Complex Topology"
- Section 36.3.4, "Setting Up the First SOA Domain"
- Section 36.3.5, "Setting Up the Second SOA Domain"
- Section 36.3.6, "Setting Up the External Portlet Domain Keystore"
- Section 36.3.7, "Setting Up the External WebCenter Domain Keystore"
- Section 36.3.8, "Command Summary for a Complex Topology"

36.3.1 Roadmap to Configuring WS-Security for a Complex Topology

The flow chart (Figure 36–12) and table (Table 36–5) in this section provide an overview of the prerequisites and tasks required to configure WS-Security for a complex multiple-domain topology.

Figure 36–12 Configuring WS-Security for a Complex Topology

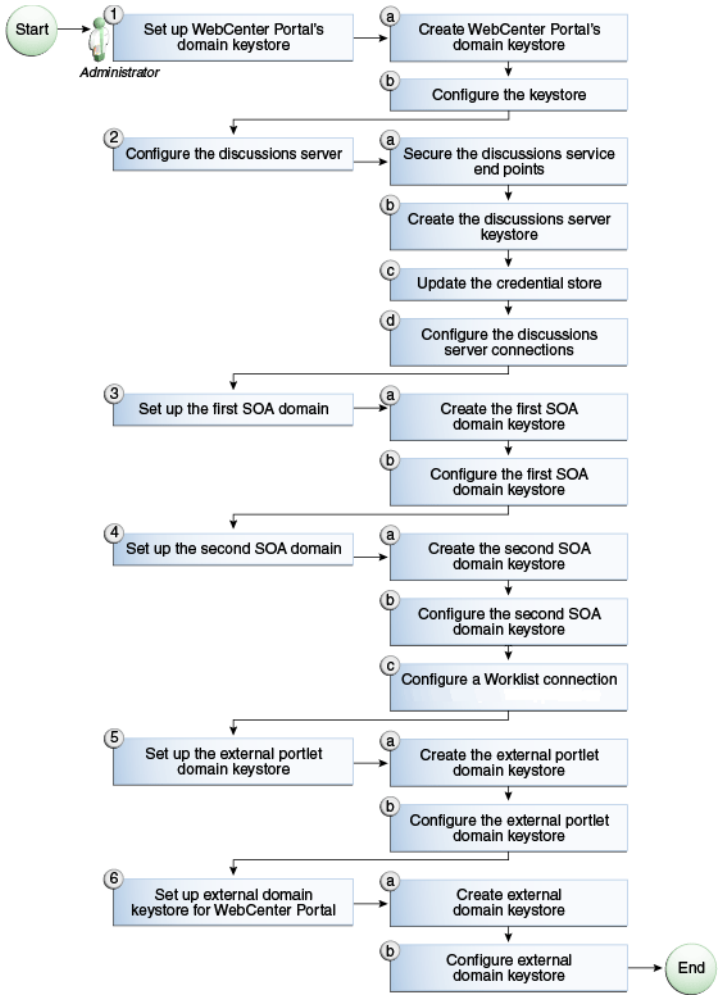


Table 36–5 shows the tasks and sub-tasks to configure WS-Security for a complex topology.

Table 36–5 Configuring WS-Security for a Complex Topology

Actor	Task	Sub-task	Notes
Administrator	**1.** Set up the WebCenter Portal domain keystore	**1.a** Create the WebCenter Portal domain keystore	
		1.b Configure the keystore	.
	2. Configure the discussions server	**2.a** Secure the discussions end points	
		2.b Create the discussions server keystore	
		2.c Update the credential store	
		2.d Configure the discussions server connections	.
	3. Set up the first SOA domain	**3.a** Create the first SOA domain keystore	
		3.b Configure the first SOA domain keystore	
	4. Set up the second SOA domain	**4.a** Create the second SOA domain keystore	

Table 36–5 (Cont.) Configuring WS-Security for a Complex Topology

Actor	Task	Sub-task	Notes
		4.b Configure the second SOA domain keystore	
		4.c Configure the worklist connection	
	5. Set up the external portlet domain keystore	**5.a** Create the external portlet domain keystore	
		5.b Configure the external portlet domain keystore	
	6. Set up the external WebCenter domain keystore	**6.a** Create the external WebCenter domain keystore	
		6.b Configure external domain keystore for the Portal Framework application	

36.3.2 Setting Up the WebCenter Portal Domain Keystores

The security credentials of WebCenter Portal, discussions server, BPEL servers (in separate domains), and WSRP producers (also in separate domains) can be retrieved and managed using a Java Keystore (JKS). A keystore is a file that provides information about available public and private keys. Keys are used for a variety of purposes, including authentication and data integrity. User certificates and the trust points needed to validate the certificates of peers are also stored securely in the wallet or keystore. See *Oracle Fusion Middleware Security and Administrator's Guide for Web Services* for information about JKS.

This section contains the following subsections:

- Section 36.3.2.1, "Creating the WebCenter Portal Domain and Framework Keystores"
- Section 36.3.2.2, "Configuring the Keystore Using WLST"
- Section 36.3.2.3, "Configuring the Keystore Using Fusion Middleware Control"

36.3.2.1 Creating the WebCenter Portal Domain and Framework Keystores

This section describes how to create the keystores and keys using a Java Keystore (JKS). JKS is the proprietary keystore format defined by Sun Microsystems. To create and manage the keys and certificates in the JKS, use the `keytool` utility that is distributed with the Java JDK 6.

To create the WebCenter Portal domain and Framework keystores:

1. Go to *`JDK_HOME`*`/jdk/bin` and open a command prompt.
2. Using keytool, generate a key pair for the `webcenter` keystore:

   ```
   keytool -genkeypair -keyalg RSA -dname "consumer_dname" -alias webcenter
   -keypass key_password -keystore keystore -storepass keystore_password -validity
   days_valid
   ```

   Where:

   - *`consumer_dname`* is the name of the consumer (for example, `cn=spaces,dc=example,dc=com`)
   - *`key_password`* is the password for the new public key, (for example, `MyPassword`)
   - *`keystore`* is the keystore name, (for example, `webcenter.jks`)

- *keystore_password* is the keystore password, (for example, MyPassword)
- *days_valid* is the number of days for which the key password is valid (for example, 1064).

Example 36–12 Generating the Keypair

```
keytool -genkeypair -keyalg RSA -dname "cn=spaces,dc=example,dc=com" -alias
webcenter -keypass MyPassword -keystore webcenter.jks -storepass MyPassword
-validity 1064
```

> **Note:** You must use the -keyalg parameter and specify RSA as its value as shown above as the default algorithm (DSA) used by keytool for generating the key is incompatible with Oracle Web Services Security Manager requirements.

3. Export the certificate containing the public key:

   ```
   keytool -exportcert -v -alias webcenter -keystore wecenter.jks
   -storepass keystore_password -rfc -file webcenter_public.cer
   ```

   Where:

   - *keystore_password* is the keystore password, (for example, MyPassword)

Example 36–13 Exporting the Certificate Containing the Public Key

```
keytool -exportcert -v -alias webcenter -keystore webcenter.jks
-storepass MyPassword -rfc -file webcenter_public.cer
```

4. Import the orakey certificate:

   ```
   keytool -importcert -alias orakey -file orakey.cer -keystore webcenter.jks
   -storepass keystore_password
   ```

   Where:

   - *keystore_password* is the keystore password, (for example, MyPassword)

Example 36–14 Importing the orakey Certificate

```
keytool -importcert -alias orakey -file orakey.cer -keystore webcenter.jks
-storepass MyPassword
```

5. Continue by configuring the keystore using either WLST, as described in Section 36.3.2.2, "Configuring the Keystore Using WLST," or using Fusion Middleware Control, as described in Section 36.3.2.3, "Configuring the Keystore Using Fusion Middleware Control."

   Table 36–6 shows the keystore contents you should wind up with after creating and configuring the keystore.

Table 36–6 WebCenter Portal Domain Keystore Contents for a Complex Topology

Key Alias	Description
webcenter	Key pair used to sign and encrypt outbound messages from WebCenter Portal. This key is used by both OWSM (portlets and worklist) and discussions.

Table 36–6 (Cont.) WebCenter Portal Domain Keystore Contents for a Complex

Key Alias	Description
orakey	Certificate containing the public key for the BPEL private key used in the SOA 1 domain. The certificate is used to encrypt outbound messages from the worklist component to SOA_Server3 in the SOA 1 domain.
soa_server3_public_key	Certificate containing the public key for the soa_server3 private key used in the SOA 2 domain. The certificate is used to encrypt outbound messages from the worklist component to BPEL Server2 in SOA 2 domain.
producer_public_key	Certificate containing public key for the producer private key used in the external portlet domain that hosts the WSRP Producer 1 application. This certificate is used to encrypt outbound messages from WebCenter Portal to WSRP Producer 1 registered in the WebCenter Portal application.
external_webcenter_custom_public_key	Certificate containing the public key for the external_webcenter_custom private key used in the external WebCenter domain that hosts the Portal Framework application that makes Web service calls to the WebCenter Portal Web service. This certificate is used to encrypt outbound messages from WebCenter Portal to Portal Framework applications in the external WebCenter domain.
owc_discussions_public	Certificate containing public key for the external owc_discussions private key used in the external discussions domain that hosts discussions. This certificate is used by WebCenter Portal and Portal Framework applications make Web service calls to the discussions Web service.

36.3.2.2 Configuring the Keystore Using WLST

After creating the WebCenter Portal domain keystore, configure the keystore service and update the credential store so that OWSM can read the keystore and keys correctly.

To configure the keystore service:

1. Go to the *<DOMAIN_HOME>*/config/fmwconfig directory, and open the file jps-config.xml in an editor.
2. Locate the <serviceInstance node for the keystore.provider Provider
3. Ensure that the webcenter.jks keystore file is copied to the *<DOMAIN_HOME>*/config/fmwconfig directory, and then specify the location as ./webcenter.jks.
4. Use the following WLST commands to update the credential store:

```
updateCred(map="oracle.wsm.security", key="keystore-csf-key", user="owsm",
password="MyPassword", desc="Keystore key")
updateCred(map="oracle.wsm.security", key="enc-csf-key", user="webcenter",
password="MyPassword", desc="Encryption key")
updateCred(map="oracle.wsm.security", key="sign-csf-key", user="webcenter",
password="MyPassword", desc="Signing key")
```

5. Restart all servers.

36.3.2.3 Configuring the Keystore Using Fusion Middleware Control

After creating the keystore, configure the keystore service and update the credential store so that OWSM can read the keystore and keys correctly.

To configure the keystore provider:

1. Open Fusion Middleware Control and log in to the WebCenter Portal domain.

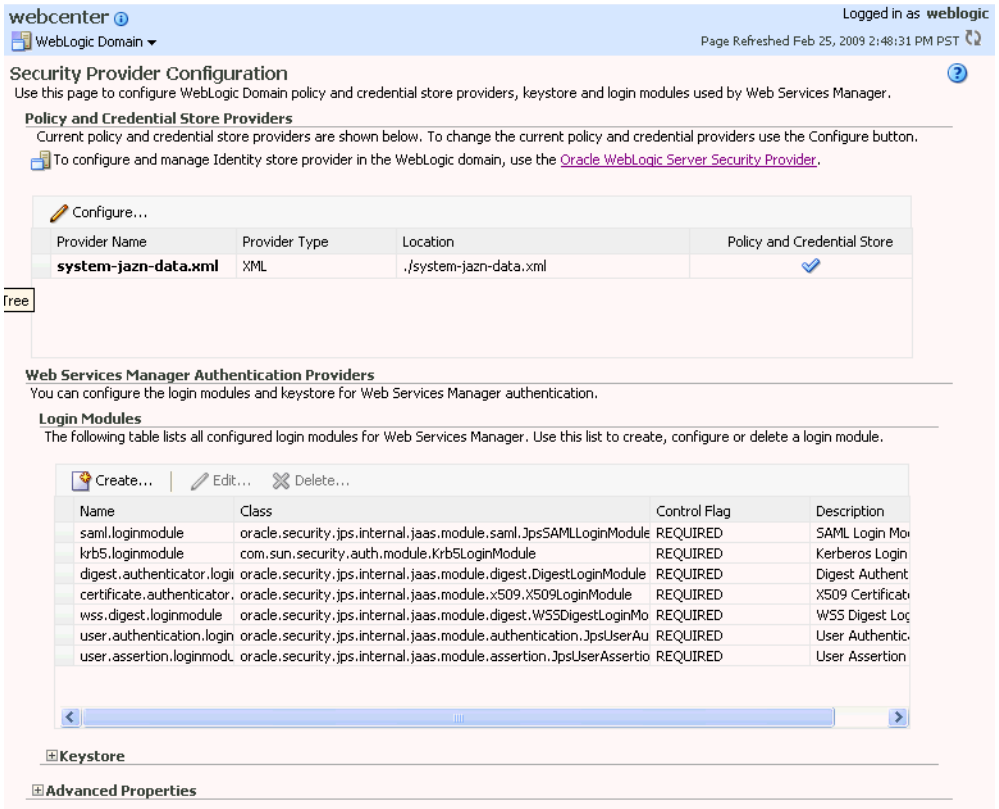
   For information on logging in to Fusion Middleware Control, see Chapter 6, "Starting Enterprise Manager Fusion Middleware Control."

2. In the Navigation pane, expand the WebLogic Domain node and click the WebCenter Portal domain (`wc_domain` by default).

3. From the WebLogic Domain menu, select **Security -> Security Provider Configuration**.

   The Security Provider Configuration page displays (see Figure 36–13).

Figure 36–13 Security Provider Configuration Page

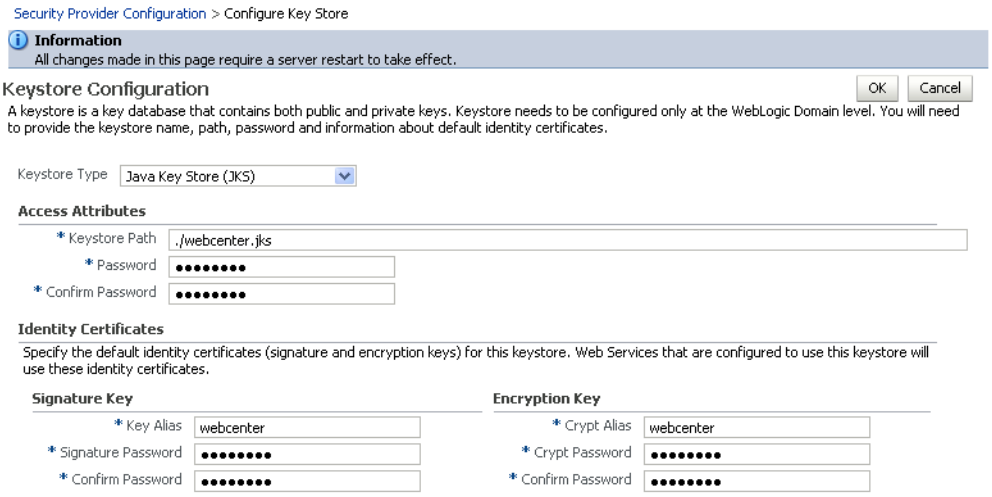


4. Expand the Keystore section on the Security Provider Configuration page.

5. Click **Configure**.

   The Keystore Configuration page displays (see Figure 36–14).

Figure 36–14 Keystore Configuration Page



6. Use the following settings to specify the location of the keystore that contains the certificate and private key, and the signature key and encryption key aliases:
   - **Keystore Path**: `./webcenter.jks`
   - **Password**: Enter and confirm the password for the keystore.
   - **Key Alias**: `webcenter`
   - **Signature Password**: Enter and confirm the password for the signature key.
   - **Crypt Alias**: `webcenter`
   - **Crypt Password**: Enter and confirm the password for the encryption key.
7. Click **OK** to save your settings.
8. Restart the Administration server for the domain.

36.3.3 Configuring the Discussions Server for a Complex Topology

In a complex topology, the discussions server is in a different domain than WebCenter Portal and consequently you will need to create and configure a keystore for the discussions server and export the certificate containing the public key and import it into the WebCenter Portal domain. For production environments you will also need to protect the discussions Web service end points with an OWSM policy and configure the discussions server connection settings. These configuration steps are described in the following subsections:

- Section 36.3.3.1, "Securing the Discussions Service End Points"
- Section 36.3.3.2, "Creating the Discussions Server Keystore"
- Section 36.3.3.3, "Updating the Credential Store"
- Section 36.3.3.4, "Configuring the Discussions Server Connection Settings"

> **Note:** Discussions-specific Web service messages sent by Portal Framework applications to the discussions server are not encrypted. For message confidentiality, the discussions server URL must be accessed over Secure Socket Layer (SSL). For more information, see Chapter 35, "Configuring SSL."

36.3.3.1 Securing the Discussions Service End Points

The discussions Web service end points require user identity to be propagated for calls originating from WebCenter Portal. Follow the steps in Section 36.1.3.2, "Securing the Discussions End Points" to secure the endpoints using either Fusion Middleware Control or WLST.

36.3.3.2 Creating the Discussions Server Keystore

This section describes how to create a keystore for the discussions server that contains the key pair used by OWSM, and export the certificate containing the public key so it can be imported into the WebCenter Portal domain.

To create the `owc_discussions` keystore:

1. Go to *`JDK_HOME`*`/jdk/bin` and open a command prompt.
2. Using keytool, generate a key pair for the `owc_discussions` keystore:

   ```
   keytool -genkeypair -keyalg RSA -dname "consumer_dname" -alias owc_discussions
   -keypass key_password -keystore keystore -storepass keystore_password -validity
   days_valid
   ```

   Where:

   - *`consumer_dname`* is the name of the consumer (for example, `cn=owc_discussions,dc=example,dc=com`)
   - *`key_password`* is the password for the new public key, (for example, `MyPassword`)
   - *`keystore`* is the keystore name, (for example, `owc_discussions.jks`)
   - *`keystore_password`* is the keystore password, (for example, `MyPassword`)
   - *`days_valid`* is the number of days for which the key password is valid (for example, `1064`).

Example 36–15 Generating the Keypair

```
keytool -genkeypair -keyalg RSA -dname "cn=owc_discussions,dc=example,dc=com"
-alias owc_discussions -keypass MyPassword -keystore owc_discussions.jks
-storepass MyPassword -validity 1064
```

> **Note:** You must use the `-keyalg` parameter and specify `RSA` as its value as shown above as the default algorithm (DSA) used by `keytool` for generating the key is incompatible with Oracle Web Services Security Manager requirements.

3. Export the certificate containing the public key:

   ```
   keytool -exportcert -v -alias owc_discussions -keystore owc_discussions.jks
   -storepass keystore_password -rfc -file owc_discussions_public.cer
   ```

Where:

- *keystore_password* is the keystore password, (for example, MyPassword)

Example 36–16 Exporting the Certificate Containing the Public Key

```
keytool -exportcert -v -alias owc_discussions -keystore owc_discussions.jks
-storepass MyPassword -rfc -file owc_discussions_public.cer
```

4. Import the webcenter_public certificate:

```
keytool -importcert -alias webcenter_public -file webcenter_public.cer
-keystore owc_discussions.jks -storepass keystore_password
```

Where:

- *keystore_password* is the keystore password, (for example, MyPassword)

Example 36–17 Importing the webcenter_public Certificate

```
keytool -importcert -alias webcenter_public -file webcenter_public.cer -keystore
owc_discussions.jks -storepass MyPassword
```

5. Import the owc_discussions_public certificate:

```
keytool -importcert -alias owc_discussions_public -file
owc_discussions_public.cer -keystore webcenter.jks -storepass keystore_password
```

Where:

- *keystore_password* is the keystore password, (for example, MyPassword)

Example 36–18 Importing the owc_discussions_public Certificate

```
keytool -importcert -alias owc_discussions_public -file owc_discussions_public.cer
-keystore webcenter.jks -storepass MyPassword
```

6. Continue by updating the credential store using WLST as described in Section 36.3.3.3, "Updating the Credential Store."

36.3.3.3 Updating the Credential Store

After creating the WebCenter Portal domain keystore, configure the keystore service and update the credential store so that OWSM can read the keystore and keys correctly.

To configure the keystore service:

1. Go to the *<DOMAIN_HOME>*/config/fmwconfig directory, and open the file jps-config.xml in an editor.
2. Locate the <serviceInstance node for the keystore.provider Provider:

```
<!-- KeyStore Service Instance -->
<serviceInstance name="keystore" provider="keystore.provider"
location="./default-keystore.jks">
<description>Default JPS Keystore Service</description>
```

3. Make sure that the webcenter.jks keystore file is copied to the *<DOMAIN_HOME>*/config/fmwconfig directory, and then specify the location as ./webcenter.jks.

```
<serviceInstance name="keystore" provider="keystore.provider"
location="./webcenter.jks">
```

```
<description>Default JPS Keystore Service</description>
```

4. Use the following WLST commands to update the credential store:

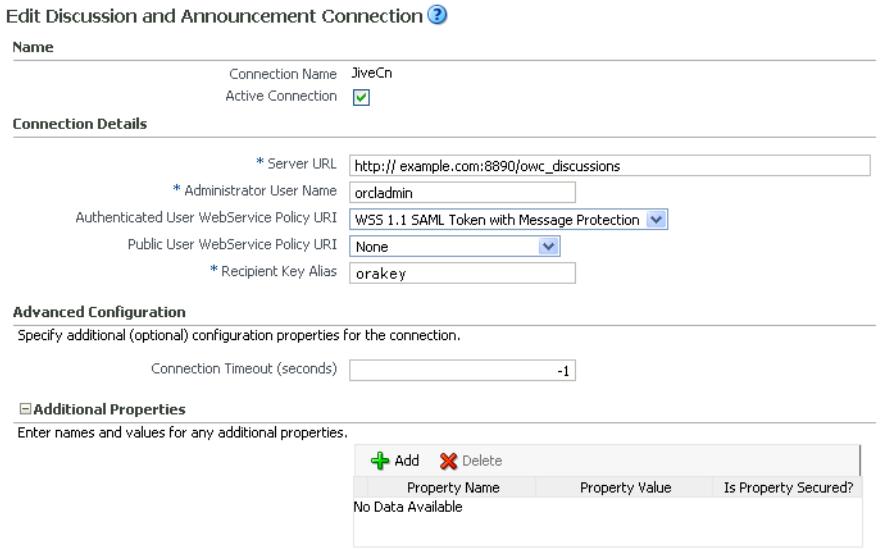
```
updateCred(map="oracle.wsm.security", key="keystore-csf-key", user="owsm",
password="MyPassword", desc="Keystore key")
updateCred(map="oracle.wsm.security", key="enc-csf-key",
user="owc_discussions", password="MyPassword", desc="Encryption key")
updateCred(map="oracle.wsm.security", key="sign-csf-key",
user="owc_discussions", password="MyPassword", desc="Signing key")
```

5. Restart all servers.

36.3.3.4 Configuring the Discussions Server Connection Settings

You must supply the WS-Security client certificate information within the discussions server connection that is configured for WebCenter Portal or your Portal Framework application, as described in Section 12.3, "Registering Discussions Servers." Figure 36–15 shows example connection detail settings for the Edit Discussions and Announcement Connection page.

Figure 36–15 Edit Discussions and Announcement Connection Page



36.3.4 Setting Up the First SOA Domain

This section describes how to set up the SOA domain keystore and contains the following subsections:

- Section 36.3.4.1, "Creating the SOA Domain Keystore"
- Section 36.3.4.2, "Configuring the Keystore Using WLST"
- Section 36.3.4.3, "Configuring the Keystore Using Fusion Middleware Control"

36.3.4.1 Creating the SOA Domain Keystore

This section describes how to create a SOA domain keystore and keys using a Java Keystore (JKS).

To create the SOA domain keystore:

1. Go to *JDK_HOME*/jdk/bin and open a command prompt.
2. Create a keystore by importing the public certificate (webcenter_public.cer) from the WebCenter Portal domain:

   ```
   keytool -importcert -alias webcenter_spaces_ws -file webcenter_public.cer
    -keystore bpel.jks -storepass keystore_password
   ```

   Where:

   - *keystore_password* is the keystore password, (for example, MyPassword)

Example 36–19 Importing the Public Certificate

```
keytool -importcert -alias webcenter_spaces_ws -file webcenter_public.cer
-keystore bpel.jks -storepass MyPassword
```

3. Using keytool, create a keypair to be used in the SOA domain for signing and encrypting messages:

   ```
   keytool -genkeypair -keyalg RSA -dname "consumer_dname" -alias bpel -keypass
   key_password -keystore bpel.jks -storepass keystore_password -validity
   days_valid
   ```

   Where:

   - *consumer_dname* is the name of the consumer (for example, cn=bpel,dc=example,dc=com)
   - *key_password* is the password for the new public key, (for example, MyPassword)
   - *keystore_password* is the keystore password, (for example, MyPassword)
   - *days_valid* is the number of days for which the key password is valid (for example, 1064).

Example 36–20 Generating the Keypair

```
keytool -genkeypair -keyalg RSA -dname "cn=bpel,dc=example,dc=com" -alias bpel
-keypass MyPassword -keystore bpel.jks -storepass MyPassword -validity 1064
```

> **Note:** You must use the -keyalg parameter and specify RSA as its value as shown above as the default algorithm (DSA) used by keytool for generating the key is incompatible with Oracle Web Services Security Manager requirements.

4. Export the certificate so it can be imported in the WebCenter Portal domain using the orakey alias:

   ```
   keytool -exportcert -v -alias bpel -keystore bpel.jks -storepass
   keystore_password -rfc -file orakey.cer
   ```

   Where:

   - *keystore_password* is the keystore password (for example, MyPassword)

Example 36–21 Exporting the Certificate Containing the Public Key

```
keytool -exportcert -v -alias bpel -keystore bpel.jks -storepass MyPassword -rfc
-file orakay.cer
```

5. Import the certificate to the WebCenter Portal domain again with a different alias (choose **Yes** when prompted whether to overwrite the existing certificate with the alias `orakey`):

   ```
   keytool -importcert -alias orakey -file orakey.cer -keystore webcenter.jks
   -storepass keystore_password
   ```

   Where:

   - *`keystore_password`* is the keystore password (for example, `MyPassword`)

Example 36–22 Importing the Certificate

```
keytool -importcert -alias orakey -file orakay.cer -keystore webcenter.jks
-storepass MyPassword
```

6. Import the certificate to the into the SOA domain:

   ```
   keytool -importcert -alias soa_server3_public_key -file
   soa_server3_public_key.cer
   -keystore webcenter.jks -storepass keystore_password
   ```

   Where:

   - *`keystore_password`* is the keystore password (for example, `MyPassword`)

Example 36–23 Importing the Certificate

```
keytool -importcert -alias soa_server3_public_key -file soa_server3_public_key.cer
-keystore webcenter.jks -storepass MyPassword
```

7. Continue by configuring the keystore using either WLST, as described in Section 36.3.4.2, "Configuring the Keystore Using WLST," or using Fusion Middleware Control as described in Section 36.3.4.3, "Configuring the Keystore Using Fusion Middleware Control."

   Table 36–7 shows the keystore contents you should wind up with after creating and configuring the SOA 1 domain keystore.

Table 36–7 SOA 1 Domain Keystore Contents for a Complex Topology

Key Alias	Description
`bpel`	Private key used to sign outbound messages from the SOA 1 domain servers. This key is used by the Worklist application deployed on the SOA 1 domain's SOA server.
`webcenter_spaces_ws`	Certificate containing the public key for the `webcenter` private key used in the WebCenter Portal domain. The certificate is used to encrypt outbound workflow messages on BPEL Server1 in the SOA 1 domain to Web service API on the WebCenter Portal domain.

36.3.4.2 Configuring the Keystore Using WLST

After creating the SOA domain keystore, configure the keystore service and update the credential store so that OWSM can read the keystore and keys correctly. You can do this using either Fusion Middleware Control, as described in Section 36.3.4.3, "Configuring the Keystore Using Fusion Middleware Control," or using WLST as described below.

To configure the keystore service:

1. Go to the *<DOMAIN_HOME>*/config/fmwconfig directory, and open the file jps-config.xml in an editor.
2. Locate the <serviceInstance node for the keystore.provider Provider
3. Ensure that the bpel.jks keystore file is copied to the *<DOMAIN_HOME>*/config/fmwconfig directory, and then specify the location as ./bpel.jks.
4. Use the following WLST commands to update the credential store:

```
updateCred(map="oracle.wsm.security", key="keystore-csf-key", user="owsm",
password="MyPassword", desc="Keystore key")
updateCred(map="oracle.wsm.security", key="enc-csf-key", user="bpel",
password="MyPassword", desc="Encryption key")
updateCred(map="oracle.wsm.security", key="sign-csf-key", user="bpel",
password="MyPassword", desc="Signing key")
```

5. Restart all servers.

36.3.4.3 Configuring the Keystore Using Fusion Middleware Control

After creating the keystore, configure the keystore service and update the credential store so that OWSM can read the keystore and keys correctly. You can do this using either WLST, as described in Section 36.3.4.2, "Configuring the Keystore Using WLST," or using Fusion Middleware Control as described below.

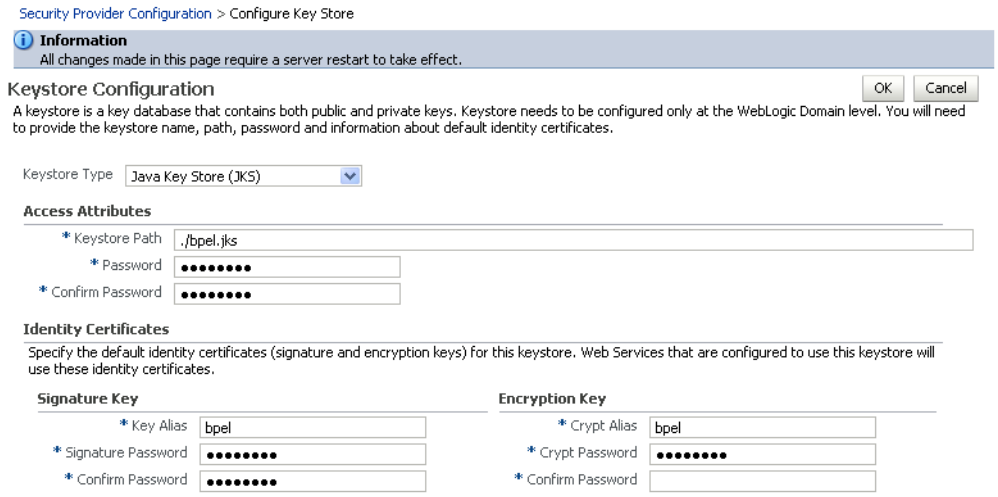
To configure the keystore provider:

1. Open Fusion Middleware Control and log in to the WebCenter Portal domain.

   For information on logging in to Fusion Middleware Control, see Chapter 6, "Starting Enterprise Manager Fusion Middleware Control."
2. In the Navigation pane, expand the WebLogic Domain node and click the SOA domain.
3. From the SOA Domain menu, select **Security -> Security Provider Configuration**.
4. Expand the Keystore section on the Security Provider Configuration page.
5. Click **Configure**.

   The Keystore Configuration page displays (see Figure 36–16).

Figure 36–16 Keystore Configuration Page



6. Use the following settings to specify the location of the keystore that contains the certificate and private key, and the signature key and encryption key aliases:
   - **Keystore Path**: `./bpel.jks`
   - **Password**: Enter and confirm the password for the keystore.
   - **Key Alias**: `bpel`
   - **Signature Password**: Enter and confirm the password for the signature key.
   - **Crypt Alias**: `bpel`
   - **Crypt Password**: Enter and confirm the password for the encryption key.
7. Click **OK** to save your settings.
8. Restart the Administration server for the domain.

36.3.5 Setting Up the Second SOA Domain

This section describes how to set up a second SOA domain keystore and contains the following subsections:

- Section 36.3.5.1, "Creating the SOA Domain Keystore"
- Section 36.3.5.2, "Configuring the Keystore Using WLST"
- Section 36.3.5.3, "Configuring the Keystore Using Fusion Middleware Control"
- Section 36.3.5.4, "Configuring the Worklist Connection for the Second SOA Server"

36.3.5.1 Creating the SOA Domain Keystore

This section describes how to create a SOA domain keystore and keys using a Java Keystore (JKS).

To create the SOA domain keystore:

1. Go to *JDK_HOME*`/jdk/bin` and open a command prompt.
2. Using keytool, create a keypair to be used in the SOA domain for signing and encrypting messages:

```
keytool -genkeypair -keyalg RSA -dname "consumer_dname" -alias soa_server3
```

```
-keypass key_password -keystore soa_server3.jks -storepass keystore_password
-validity days_valid
```

Where:

- *consumer_dname* is the name of the consumer (for example, `cn=soa_server3,dc=example,dc=com`)
- *key_password* is the password for the new public key, (for example, `MyPassword`)
- *keystore_password* is the keystore password, (for example, `MyPassword`)
- *days_valid* is the number of days for which the key password is valid (for example, `1064`).

Example 36–24 Generating the Keypair

```
keytool -genkeypair -keyalg RSA -dname "cn=soa_server3,dc=example,dc=com" -alias
soa_server3 -keypass MyPassword -keystore soa_server3.jks -storepass MyPassword
-validity 1064
```

Note: You must use the `-keyalg` parameter and specify `RSA` as its value as shown above as the default algorithm (DSA) used by `keytool` for generating the key is incompatible with Oracle Web Services Security Manager requirements.

3. Export the certificate so it can be imported in the WebCenter Portal domain using the `orakey` alias:

```
keytool -exportcert -v -alias soa_server3 -keystore soa_server3.jks
 -storepass keystore_password -rfc -file soa_server3_public_key.cer
```

Where:

- *keystore_password* is the keystore password, (for example, `MyPassword`)

Example 36–25 Exporting the Certificate Containing the Public Key

```
keytool -exportcert -v -alias soa_server3 -keystore soa_server3.jks
-storepass MyPassword -rfc -file soa_server3_public_key.cer
```

4. Import the certificate to the WebCenter Portal domain with a different alias (choose **Yes** when prompted whether to overwrite the existing certificate with the alias `soa_server3_public_key`):

```
keytool -importcert -alias soa_server3_public_key -file soa_server3_public_
key.cer  -keystore webcenter.jks -storepass keystore_password
```

Where:

- *keystore_password* is the keystore password (for example, `MyPassword`)

Example 36–26 Importing the Certificate

```
keytool -importcert -alias soa_server3_public_key -file soa_server3_public_key.cer
-keystore webcenter.jks -storepass MyPassword
```

5. Import the `soa_server3_public_key` certificate:

```
keytool -importcert -alias soa_server3_public_key -file
```

```
soa_server3_public_key.cer -keystore webcenter.jks -storepass keystore_password
```

Where:

- *keystore_password* is the keystore password, (for example, `MyPassword`)

Example 36–27 Importing the soa_server3_public_key Certificate

```
keytool -importcert -alias soa_server3_public_key -file soa_server3_public_key.cer
-keystore webcenter.jks -storepass MyPassword
```

6. Import the `producer_public_key` certificate:

```
keytool -importcert -alias producer_public_key -file producer_public_key.cer
-keystore webcenter.jks -storepass keystore_password
```

Where:

- *keystore_password* is the keystore password, (for example, `MyPassword`)

Example 36–28 Importing the producer_public_key Certificate

```
keytool -importcert -alias producer_public_key -file producer_public_key.cer
-keystore webcenter.jks -storepass MyPassword
```

7. Import the `external_webcenter_custom_public_key` certificate:

```
keytool -importcert -alias external_webcenter_custom_public_key -file
external_webcenter_custom_public_key.cer -keystore webcenter.jks -storepass
keystore_password
```

Where:

- *keystore_password* is the keystore password, (for example, `MyPassword`)

Example 36–29 Importing the external_webcenter_custom_public_key Certificate

```
keytool -importcert -alias external_webcenter_custom_public_key -file
external_webcenter_custom_public_key.cer -keystore webcenter.jks -storepass
MyPassword
```

8. Continue by configuring the keystore using either WLST, as described in Section 36.3.5.2, "Configuring the Keystore Using WLST," or using Fusion Middleware Control as described in Section 36.3.5.3, "Configuring the Keystore Using Fusion Middleware Control."

   Table 36–8 shows the keystore contents you should wind up with after creating and configuring the SOA 2 domain keystore.

Table 36–8 SOA 2 Domain Keystore Contents for a Complex Topology

Key Alias	Description
`webcenter`	Key pair used to sign and encrypt outbound messages from WebCenter Portal. This key is used by both OWSM (portlets and worklist) and discussions.
`orakey`	Certificate containing the public key for the `BPEL` private key used in the SOA 1 domain. The certificate is used to encrypt outbound messages from the worklist component to SOA_Server3 in the SOA 1 domain.

Table 36–8 (Cont.) SOA 2 Domain Keystore Contents for a Complex Topology

Key Alias	Description
`soa_server3_public_key`	Certificate containing the public key for the soa_server3 private key used in the SOA 2 domain. The certificate is used to encrypt outbound messages from the worklist component to BPEL Server2 in SOA 2 domain.
`producer_public_key`	Certificate containing public key for the producer private key used in the external portlet domain that hosts the WSRP Producer 1 application. This certificate is used to encrypt outbound messages from WebCenter Portal to WSRP Producer 1 registered in the WebCenter Portal application.
`external_webcenter_custom_public_key`	Certificate containing the public key for the external_webcenter_custom private key used in the external WebCenter domain that hosts the Portal Framework application that makes Web service calls to the WebCenter Portal Web service. This certificate is used to encrypt outbound messages from WebCenter Portal to Portal Framework applications in the external WebCenter domain.

36.3.5.2 Configuring the Keystore Using WLST

After creating the second SOA domain keystore, configure the keystore service and update the credential store so that OWSM can read the keystore and keys correctly. You can do this using either Fusion Middleware Control, as described in Section 36.3.5.3, "Configuring the Keystore Using Fusion Middleware Control," or using WLST as described below.

To configure the keystore service:

1. Go to the *<DOMAIN_HOME>*`/config/fmwconfig` directory, and open the file `jps-config.xml` in an editor.
2. Locate the `<serviceInstance` node for the keystore.provider `Provider`
3. Ensure that the `soa_server3.jks` keystore file is copied to the *<DOMAIN_HOME>*`/config/fmwconfig` directory, and then specify the location as `./soa_server3.jks`.
4. Use the following WLST commands to update the credential store:

```
updateCred(map="oracle.wsm.security", key="keystore-csf-key", user="owsm",
password="MyPassword", desc="Keystore key")
updateCred(map="oracle.wsm.security", key="enc-csf-key", user="soa_server3",
password="MyPassword", desc="Encryption key")
updateCred(map="oracle.wsm.security", key="sign-csf-key", user="soa_server3",
password="MyPassword", desc="Signing key")
```

5. Restart all servers.

36.3.5.3 Configuring the Keystore Using Fusion Middleware Control

After creating the keystore, configure the keystore service and update the credential store so that OWSM can read the keystore and keys correctly. You can do this using either WLST, as described in Section 36.3.5.2, "Configuring the Keystore Using WLST," or using Fusion Middleware Control as described below.

To configure the keystore provider:

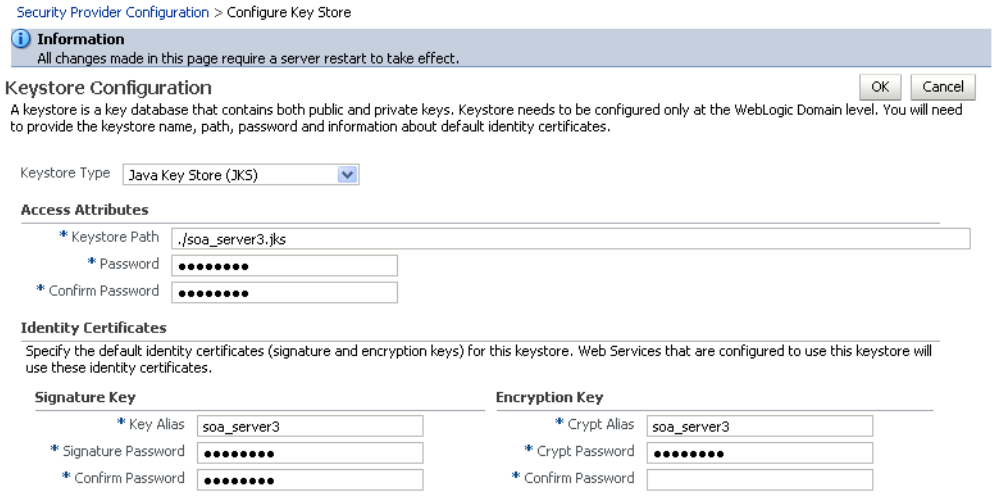
1. Open Fusion Middleware Control and log in to the WebCenter Portal domain.

For information on logging in to Fusion Middleware Control, see Chapter 6, "Starting Enterprise Manager Fusion Middleware Control."

2. In the Navigation pane, expand the WebLogic Domain node and click the SOA domain.
3. From the SOA Domain menu, select **Security -> Security Provider Configuration**.
4. Expand the Keystore section on the Security Provider Configuration page.
5. Click **Configure**.

   The Keystore Configuration page displays (see Figure 36–17).

Figure 36–17 Keystore Configuration Page



6. Use the following settings to specify the location of the keystore that contains the certificate and private key, and the signature key and encryption key aliases:
   - **Keystore Path**: `./soa_server3.jks`
   - **Password**: Enter and confirm the password for the keystore.
   - **Key Alias**: `soa_server3`
   - **Signature Password**: Enter and confirm the password for the signature key.
   - **Crypt Alias**: `soa_server3`
   - **Crypt Password**: Enter and confirm the password for the encryption key.
7. Click **OK** to save your settings.
8. Restart the Administration server for the domain.

36.3.5.4 Configuring the Worklist Connection for the Second SOA Server

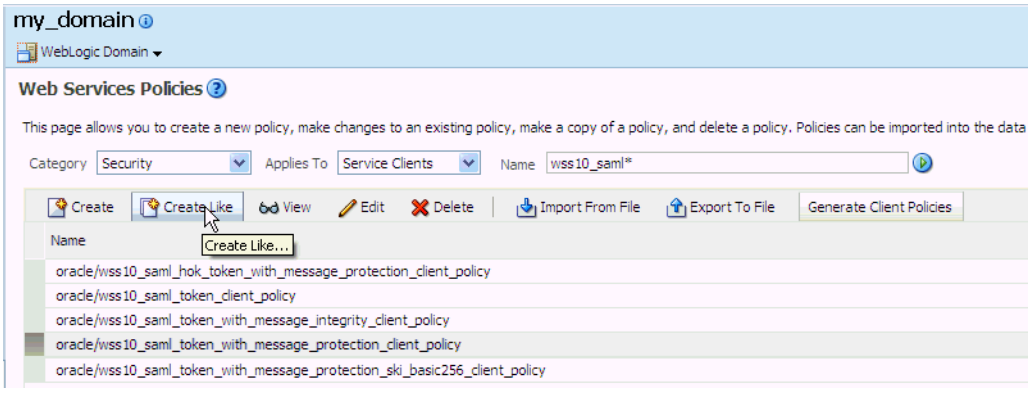
Ordinarily, the worklist connection uses the `oracle/wss10_saml_token_with_message_protection_client_policy` policy to secure outbound SOAP messages to SOA Server. However, in a complex deployment where the WebCenter Portal domain uses two or more worklist connections simultaneously we need to create an additional OWSM policy and configure it so that the recipient key alias matches the alias of the certificate of the intended SOA server on the WebCenter Portal side.

Follow the steps below to use multiple worklist connections simultaneously:

1. Export the certificate from the external SOA domain and import it into the WebCenter Portal domain under a new alias (`soa_server3_key` in the following example).
2. Use Fusion Middleware Control to create a new OWSM policy, and override the recipient key alias to use the same alias as above.
   a. In Fusion Middleware Control, from the WebLogic domain menu select **Web Services -> Policies**.

      The Web Services Policies page displays (see Figure 36–18).

Figure 36–18 Web Services Policies Page



   b. Select a client policy to use as a base for creating the new policy and click **Create Like**.

      The Create Policy page displays (see Figure 36–19).

Figure 36–19 Create Policy Page

   c. Enter a name for the new policy (for example, `oracle_wss10_saml_token_with_message_protection_client_policy_soa_server3`) and click **Save**.

      The new policy should now be listed on the Web Services Policies page.

   d. From the Web Services Policy page, select the new policy and click **Edit**.
   e. On the Edit Policy page, open the Configuration tab and click **Edit**.
   f. Override the recipient key alias with the value `soa_server3_key` and click **Save**.
3. Create the BPEL connection to set the security policy to the policy created above using the following WLST command:

```
setBPELConnection(appName='webcenter',
```

```
name='WebCenter-Worklist-SOAServer3',url='<your_url>',
policy='oracle/wss10_saml_token_with_message_protection_client_policy_soa_serve
r3')
```

36.3.6 Setting Up the External Portlet Domain Keystore

This section describes how to set up the keystore for the external portlet domain used by one of the WSRP producers for this complex topology.

This section contains the following subsections:

- Section 36.3.6.1, "Creating the External Portlet Domain Keystore"
- Section 36.3.6.2, "Configuring the Keystore Using WLST"
- Section 36.3.6.3, "Configuring the Keystore Using Fusion Middleware Control"

36.3.6.1 Creating the External Portlet Domain Keystore

To create the external portlet domain keystore:

1. Go to *JDK_HOME*/jdk/bin and open a command prompt.
2. Using keytool, generate the keystore by importing the WebCenter Portal domain's public certificate:

   ```
   keytool -importcert -alias webcenter_public -file webcenter_public.cer
   -keystore producer.jks -storepass keystore_password
   ```

   Where:

   - *keystore_password* is the keystore password

Example 36–30 Importing the Certificate

```
keytool -importcert -alias webcenter_public -file webcenter_public.cer -keystore
producer.jks -storepass MyPassword
```

3. Using keytool, generate a key pair:

   ```
   keytool -genkeypair -keyalg RSA -dname "consumer_dname" -alias producer
    -keypass key_password -keystore producer.jks -storepass keystore_password
    -validity days_valid
   ```

   Where:

   - *consumer_dname* is the name of the consumer (for example, cn=producer,dc=example,dc=com)
   - *key_password* is the password for the new public key, (for example, MyPassword)
   - *keystore* is the keystore name, (for example, webcenter.jks)
   - *keystore_password* is the keystore password, (for example, MyPassword)
   - *days_valid* is the number of days for which the key password is valid (for example, 1064).

Example 36–31 Generating the Keypair

```
keytool -genkeypair -keyalg RSA -dname "cn=producer,dc=example,dc=com" -alias
producer -keypass MyPassword -keystore producer.jks -storepass MyPassword
 -validity 1064
```

Note: You must use the `-keyalg` parameter and specify `RSA` as its value as shown above as the default algorithm (DSA) used by `keytool` for generating the key is incompatible with Oracle Web Services Security Manager requirements.

4. Export the certificate containing the public key so that it can be imported into the WebCenter Portal domain's keystore:

   ```
   keytool -exportcert -v -alias producer -keystore producer.jks -storepass
   keystore_password -rfc -file producer_public_key.cer
   ```

   Where:

   - *keystore_password* is the keystore password, (for example, `MyPassword`)

Example 36–32 Exporting the Certificate Containing the Public Key

```
keytool -exportcert -v -alias producer -keystore producer.jks -storepass
MyPassword
-rfc -file producer_public_key.cer
```

5. Import the certificate to the WebCenter Portal domain with a different alias (choose **Yes** when prompted whether to overwrite the existing certificate with the alias `producer_public_key`):

   ```
   keytool -importcert -alias producer_public_key -file producer_public_key.cer
   -keystore webcenter.jks -storepass keystore_password
   ```

   Where:

   - *keystore_password* is the keystore password (for example, `MyPassword`)

Example 36–33 Importing the Certificate

```
keytool -importcert -alias producer_public_key -file producer_public_key.cer
-keystore webcenter.jks -storepass MyPassword
```

6. Continue by configuring the keystore using either WLST as described in Section 36.3.6.2, "Configuring the Keystore Using WLST," or using Fusion Middleware Control as described in Section 36.3.6.3, "Configuring the Keystore Using Fusion Middleware Control."

36.3.6.2 Configuring the Keystore Using WLST

After creating the external portlet domain keystore, configure the keystore service and update the credential store so that OWSM can read the keystore and keys correctly. You can do this using either Fusion Middleware Control, as described in Section 36.3.6.3, "Configuring the Keystore Using Fusion Middleware Control," or using WLST as described below.

To configure the keystore service:

1. Go to the *<DOMAIN_HOME>*`/config/fmwconfig` directory, and open the file `jps-config.xml` in an editor.
2. Locate the `<serviceInstance` node for the keystore.provider `Provider`
3. Ensure that the `producer.jks` keystore file is copied to the *<DOMAIN_HOME>*`/config/fmwconfig` directory, and then specify the location as `./producer.jks`.

4. Use the following WLST commands to update the credential store:

```
updateCred(map="oracle.wsm.security", key="keystore-csf-key", user="owsm",
password="MyPassword", desc="Keystore key")
updateCred(map="oracle.wsm.security", key="enc-csf-key", user="producer",
password="MyPassword", desc="Encryption key")
updateCred(map="oracle.wsm.security", key="sign-csf-key", user="producer",
password="MyPassword", desc="Signing key")
```

5. Restart all servers.

36.3.6.3 Configuring the Keystore Using Fusion Middleware Control

After creating the keystore, configure the keystore service and update the credential store so that OWSM can read the keystore and keys correctly. You can do this using either WLST, as described in Section 36.3.6.2, "Configuring the Keystore Using WLST," or using Fusion Middleware Control as described below.

To configure the keystore provider:

1. Open Fusion Middleware Control and log in to the WebCenter Portal domain.

   For information on logging in to Fusion Middleware Control, see Chapter 6, "Starting Enterprise Manager Fusion Middleware Control."

2. In the Navigation pane, expand the WebLogic Domain node and click the WebCenter Portal domain (`wc_domain` by default).

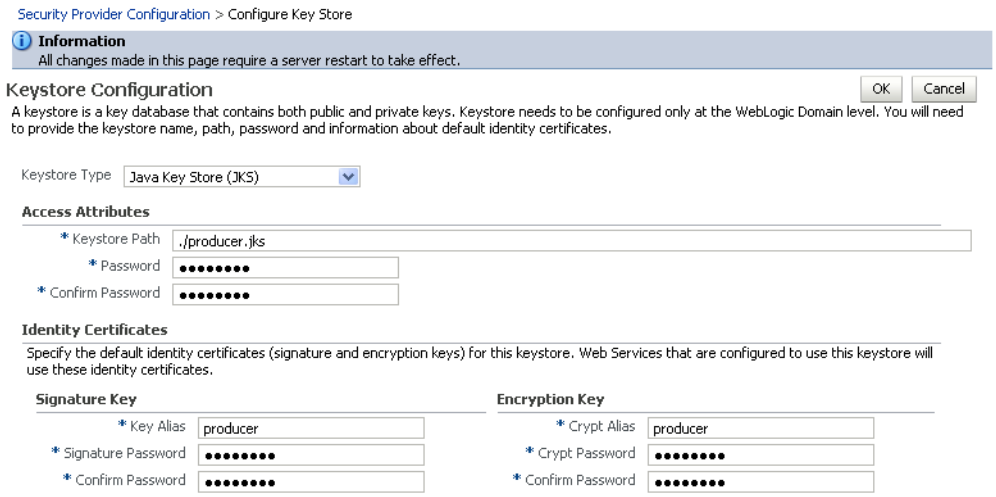
3. From the WebLogic Domain menu, select **Security -> Security Provider Configuration**.

4. Expand the Keystore section on the Security Provider Configuration page.

5. Click **Configure**.

   The Keystore Configuration page displays (see Figure 36–20).

Figure 36–20 Keystore Configuration Page



6. Use the following settings to specify the location of the keystore that contains the certificate and private key, and the signature key and encryption key aliases:

   - **Keystore Path**: `./producer.jks`
   - **Password**: Enter and confirm the password for the keystore.

- **Key Alias**: `producer`
- **Signature Password**: Enter and confirm the password for the signature key.
- **Crypt Alias**: `producer`
- **Crypt Password**: Enter and confirm the password for the encryption key.

7. Click **OK** to save your settings.
8. Restart the Administration server for the domain.

36.3.7 Setting Up the External WebCenter Domain Keystore

This section describes how to set up an external WebCenter domain used by a Portal Framework application making WebCenter Portal Web service calls.

This section contains the following subsections:

- Section 36.3.7.1, "Creating the External WebCenter Domain Keystore"
- Section 36.3.7.2, "Configuring the Keystore Using WLST"
- Section 36.3.7.3, "Configuring the Keystore Using Fusion Middleware Control"

36.3.7.1 Creating the External WebCenter Domain Keystore

To create the external WebCenter domain keystore:

1. Go to *JDK_HOME*`/jdk/bin` and open a command prompt.
2. Using keytool, generate the keystore by importing the WebCenter domain's public certificate:

```
keytool -importcert -alias webcenter_public -file webcenter_public.cer
 -keystore external_webcenter_custom.jks -storepass keystore_password
```

Where:

- *keystore_password* is the keystore password

Example 36–34 Importing the Certificate

```
keytool -importcert -alias webcenter_public -file webcenter_public.cer -keystore
external_webcenter_custom.jks -storepass MyPassword
```

3. Using keytool, generate a key pair:

```
keytool -genkeypair -keyalg RSA -dname "consumer_dname" -alias
external_webcenter_custom -keypass key_password -keystore
external_webcenter_custom.jks -storepass keystore_password -validity days_valid
```

Where:

- *consumer_dname* is the name of the consumer (for example, `cn=external_webcenter_custom,dc=example,dc=com`)
- *key_password* is the password for the new public key, (for example, `MyPassword`)
- *keystore_password* is the keystore password, (for example, `MyPassword`)
- *days_valid* is the number of days for which the key password is valid (for example, `1064`).

Example 36–35 Generating the Keypair

```
keytool -genkeypair -keyalg RSA -dname "cn=external_webcenter_custom,
dc=example,dc=com" -alias external_webcenter_custom -keypass MyPassword
-keystore external_webcenter_custom.jks -storepass MyPassword -validity 1064
```

> **Note:** You must use the `-keyalg` parameter and specify `RSA` as its value as shown above as the default algorithm (DSA) used by `keytool` for generating the key is incompatible with Oracle Web Services Security Manager requirements.

4. Export the certificate containing the public key so that it can be imported into the WebCenter Portal domain's keystore:

   ```
   keytool -exportcert -v -alias external_webcenter_custom -keystore external_
   webcenter_custom.jks -storepass keystore_password -rfc -file external_
   webcenter_custom_public_key.cer
   ```

   Where:

   - *`keystore_password`* is the keystore password, (for example, `MyPassword`)

Example 36–36 Exporting the Certificate Containing the Public Key

```
keytool -exportcert -v -alias external_webcenter_custom -keystore external_
webcenter_custom.jks -storepass MyPassword -rfc -file external_webcenter_custom_
public_key.cer
```

5. Import the certificate to the WebCenter Portal domain with a different alias (choose **Yes** when prompted whether to overwrite the existing certificate with the alias external_webcenter_custom_public_key):

   ```
   keytool -importcert -alias external_webcenter_custom_public_key -file
   external_webcenter_custom_public_key.cer -keystore webcenter.jks
   -storepass keystore_password
   ```

   Where:

   - *`keystore_password`* is the keystore password (for example, `MyPassword`)

Example 36–37 Importing the Certificate

```
keytool -importcert -alias external_webcenter_custom_public_key -file external_
webcenter_custom_public_key.cer -keystore webcenter.jks -storepass MyPassword
```

6. Continue by configuring the keystore using either WLST as described in Section 36.3.7.2, "Configuring the Keystore Using WLST," or using Fusion Middleware Control as described in Section 36.3.7.3, "Configuring the Keystore Using Fusion Middleware Control."

36.3.7.2 Configuring the Keystore Using WLST

After creating the external WebCenter domain keystore, configure the keystore service and update the credential store so that OWSM can read the keystore and keys correctly. You can do this using either Fusion Middleware Control, as described in Section 36.3.7.3, "Configuring the Keystore Using Fusion Middleware Control," or using WLST as described below.

To configure the keystore service:

1. Go to the *<DOMAIN_HOME>*/config/fmwconfig directory, and open the file jps-config.xml in an editor.
2. Locate the <serviceInstance node for the keystore.provider Provider
3. Ensure that the webcenter.jks keystore file is copied to the *<DOMAIN_HOME>*/config/fmwconfig directory, and then specify the location as ./webcenter.jks.
4. Use the following WLST commands to update the credential store:

```
updateCred(map="oracle.wsm.security", key="keystore-csf-key", user="owsm",
password="MyPassword", desc="Keystore key")
updateCred(map="oracle.wsm.security", key="enc-csf-key",
user="external_webcenter_custom",
password="MyPassword", desc="Encryption key")
updateCred(map="oracle.wsm.security", key="sign-csf-key",
user="external_webcenter_custom", password="MyPassword", desc="Signing key")
```

5. Restart all servers.

36.3.7.3 Configuring the Keystore Using Fusion Middleware Control

After creating the keystore, configure the keystore service and update the credential store so that OWSM can read the keystore and keys correctly. You can do this using either WLST, as described in Section 36.3.7.2, "Configuring the Keystore Using WLST," or using Fusion Middleware Control as described below.

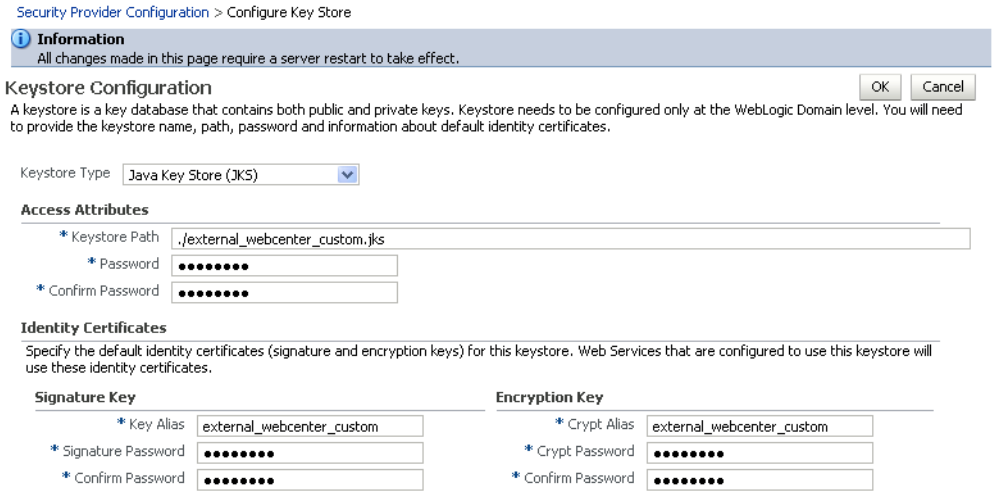
To configure the keystore provider:

1. Open Fusion Middleware Control and log into the WebCenter Portal domain.

   For information on logging in to Fusion Middleware Control, see Chapter 6, "Starting Enterprise Manager Fusion Middleware Control."
2. In the Navigation pane, expand the WebLogic Domain node and click the WebCenter Portal domain (wc_domain by default).
3. From the WebLogic Domain menu, select **Security -> Security Provider Configuration**.
4. Expand the Keystore section on the Security Provider Configuration page.
5. Click **Configure**.

   The Keystore Configuration page displays (see Figure 36–21).

Figure 36–21 Keystore Configuration Page



6. Use the following settings to specify the location of the keystore that contains the certificate and private key, and the signature key and encryption key aliases:
   - **Keystore Path**: `./external_webcenter_custom.jks`
   - **Password**: Enter and confirm the password for the keystore.
   - **Key Alias**: `external_webcenter_custom`
   - **Signature Password**: Enter and confirm the password for the signature key.
   - **Crypt Alias**: `external_webcenter_custom`
   - **Crypt Password**: Enter and confirm the password for the encryption key.
7. Click **OK** to save your settings.
8. Restart the Administration server for the domain.

36.3.8 Command Summary for a Complex Topology

Use the following command summary to quickly configure the keystore and DF properties for a complex topology.

Generate the Keystore

Use the following `keytool` commands to generate the keystore, replacing the values in bold with those for your local environment:

```
keytool -genkeypair -keyalg RSA -dname "cn=spaces,dc=example,dc=com" -alias
webcenter  -keypass MyPassword -keystore webcenter.jks -storepass MyPassword
-validity 1064

keytool -exportcert -v -alias webcenter -keystore webcenter.jks -storepass
MyPassword -rfc -file webcenter_public.cer

keytool -importcert -alias df_webcenter_public -file webcenter_public.cer
-keystore owc_discussions.jks -storepass MyPassword
```

When prompted to trust the certificate, say `yes`.

```
keytool -importcert -alias webcenter_spaces_ws -file webcenter_public.cer
-keystore bpel.jks -storepass MyPassword
```

When prompted to trust the certificate, say `yes`.

```
keytool -genkeypair -keyalg RSA -dname "cn=bpel,dc=example,dc=com" -alias bpel
-keypass MyPassword -keystore bpel.jks

keytool -exportcert -v -alias bpel -keystore bpel.jks -storepass MyPassword -rfc
-file orakay.cer

keytool -importcert -alias orakey -file orakay.cer -keystore webcenter.jks
-storepass MyPassword
```

When prompted to trust the certificate, say `yes`.

```
keytool -genkeypair -keyalg RSA -dname "cn=soa_server3,dc=example,dc=com" -alias
 soa_server3 -keypass MyPassword -keystore soa_server3.jks -storepass MyPassword
-validity 1024

keytool -exportcert -v -alias soa_server3 -keystore soa_server3.jks -storepass
MyPassword -rfc -file soa_server3_public_key.cer

keytool -importcert -alias soa_server3_public_key -file soa_server3_public_key.cer
-keystore webcenter.jks -storepass MyPassword
```

When prompted to trust the certificate, say `yes`.

```
keytool -importcert -alias webcenter_public -file webcenter_public.cer -keystore
producer.jks -storepass MyPassword
```

When prompted to trust the certificate, say `yes`.

```
keytool -genkeypair -keyalg RSA -dname "cn=producer,dc=example,dc=com" -alias
producer -keypass MyPassword -keystore producer.jks -storepass MyPassword
-validity 1024

keytool -exportcert -v -alias producer -keystore producer.jks
-storepass MyPassword -rfc -file producer_public_key.cer

keytool -importcert -alias webcenter_public -file webcenter_public.cer
-keystore external_webcenter_custom.jks -storepass MyPassword
```

When prompted to trust the certificate, say `yes`.

```
keytool -genkeypair -keyalg RSA -dname
"cn=external_webcenter_custom,dc=example,dc=com"
-alias external_webcenter_custom -keypass MyPassword -keystore
external_webcenter_custom.jks
-storepass MyPassword -validity 1024

keytool -exportcert -v -alias external_webcenter_custom -keystore
external_webcenter_custom.jks -storepass MyPassword -rfc -file
external_webcenter_custom_public_key.cer

keytool -importcert -alias producer_public_key -file producer_public_key.cer
-keystore webcenter.jks -storepass MyPassword
```

When prompted to trust the certificate, say `yes`.

```
keytool -importcert -alias external_webcenter_custom_public_key -file
external_webcenter_custom_public_key.cer -keystore webcenter.jks -storepass
```

```
MyPassword
```

When prompted to trust the certificate, say `yes`.

Copy `webcenter.jks` to your `domain_home/config/fmwconfig` directory, `bpel.jks` to your `SOA1_domain_home/config/fmwconfig` directory, `soa_server3.jks` to your `SOA_2_domain_home/config/fmwconfig` directory, `producer.jks` to your `External_Portlet_domain_home/config/fmwconfig` directory, and `external_webcenter_custom.jks` to your `External_WebCenter_domain_home/config/fmwconfig` directory.

Configure the WebCenter Portal Domain Keystore

Follow the steps below to configure the service instance reference for the WebCenter Portal domain:

1. Navigate to the `<DOMAIN_HOME>/config/fmwconfig` directory.
2. Copy `webcenter.jks` to the `<DOMAIN_HOME>/config/fmwconfig` directory if you haven't already done so.
3. Open `jps-config.xml` in an editor.
4. Locate `<serviceInstance node for keystore.provider Provider`.
5. Specify the location as `./webcenter.jks`.
6. Using WLST, connect to the WebCenter Portal domain as an administrator and run the following commands:

```
updateCred(map="oracle.wsm.security", key="keystore-csf-key", user="owsm",
password="MyPassword", desc="Keystore key")
updateCred(map="oracle.wsm.security", key="enc-csf-key", user="webcenter",
password="MyPassword", desc="Encryption key")
updateCred(map="oracle.wsm.security", key="sign-csf-key", user="webcenter",
password="MyPassword", desc="Signing key")
```

Configure the External Discussions Server Domain Keystore

Follow the steps below to configure the service instance reference for the discussions server:

1. Navigate to the `<DOMAIN_HOME>/config/fmwconfig` directory.
2. Copy `webcenter.jks` to the `<DOMAIN_HOME>/config/fmwconfig` directory if you haven't already done so.
3. Open `jps-config.xml` in an editor.
4. Locate `<serviceInstance node for keystore.provider Provider`.
5. Specify the location as `./owc_discussions.jks`.
6. Using WLST, connect to the WebCenter Portal domain as an administrator and run the following commands:

```
updateCred(map="oracle.wsm.security", key="keystore-csf-key", user="owsm",
password="MyPassword", desc="Keystore key")
updateCred(map="oracle.wsm.security", key="enc-csf-key",
user="owc_discussions", password="MyPassword", desc="Encryption key")
updateCred(map="oracle.wsm.security", key="sign-csf-key",
user="owc_discussions", password="MyPassword", desc="Signing key")
```

Configure the SOA1 Domain Keystore

Follow the steps below to configure the service instance reference for the SOA1 domain:

1. Navigate to the `<DOMAIN_HOME>/config/fmwconfig` directory.
2. Copy `bpel.jks` to the `<DOMAIN_HOME>/config/fmwconfig` directory if you haven't done already done so.
3. Open `jps-config.xml` in an editor.
4. Locate `<serviceInstance node for keystore.provider Provider`.
5. Specify the location as `./bpel.jks`.
6. Using WLST, connect to the SOA1 domain as an admin user and run the following commands:

```
updateCred(map="oracle.wsm.security", key="keystore-csf-key", user="owsm",
password="MyPassword", desc="Keystore key")
updateCred(map="oracle.wsm.security", key="enc-csf-key", user="bpel",
password="MyPassword", desc="Encryption key")
updateCred(map="oracle.wsm.security", key="sign-csf-key", user="bpel",
password="MyPassword", desc="Signing key")
```

Configure the SOA2 Domain Keystore

Follow the steps below to configure the service instance reference for the SOA2 domain:

1. Navigate to the `<DOMAIN_HOME>/config/fmwconfig` directory.
2. Copy `soa_server3.jks` to the `<DOMAIN_HOME>/config/fmwconfig` directory if you haven't done already done so.
3. Open `jps-config.xml` in an editor.
4. Locate `<serviceInstance node for keystore.provider Provider`.
5. Specify the location as `./soa_server3.jks`.
6. Using WLST, connect to the SOA2 domain as an admin user and run the following commands:

```
updateCred(map="oracle.wsm.security", key="keystore-csf-key", user="owsm",
password="MyPassword", desc="Keystore key")
updateCred(map="oracle.wsm.security", key="enc-csf-key", user="soa_server3",
password="MyPassword", desc="Encryption key")
updateCred(map="oracle.wsm.security", key="sign-csf-key", user="soa_server3",
password="MyPassword", desc="Signing key")
```

Configure the External Portlet Producer Domain Keystore

Follow the steps below to configure the service instance reference for the External Portlet Producer and External WebCenter domain keystores:

1. Navigate to the `<DOMAIN_HOME>/config/fmwconfig` directory of the External Portlet Producer domain.
2. Copy `producer.jks` to the `<DOMAIN_HOME>/config/fmwconfig` directory if you haven't done already done so.
3. Open `jps-config.xml` in an editor.
4. Locate `<serviceInstance node for keystore.provider Provider`.
5. Specify the location as `./producer.jks`.

6. Using WLST, connect to the External Portlet Producer domain as an administrator and run the following commands:

```
updateCred(map="oracle.wsm.security", key="keystore-csf-key", user="owsm",
password="MyPassword", desc="Keystore key")
updateCred(map="oracle.wsm.security", key="enc-csf-key", user="producer",
password="MyPassword", desc="Encryption key")
updateCred(map="oracle.wsm.security", key="sign-csf-key", user="producer",
password="MyPassword", desc="Signing key")
```

7. Navigate to the `<DOMAIN_HOME>/config/fmwconfig` directory of the External WebCenter domain.
8. Copy `producer.jks` to the `<DOMAIN_HOME>/config/fmwconfig` directory if you haven't done already done so.
9. Open `jps-config.xml` in an editor.
10. Locate `<serviceInstance node for keystore.provider Provider`.
11. Specify the location as `./external_webcenter_custom.jks`.
12. Using WLST, connect to the External Portlet Producer domain as an administrator and run the following commands:

```
updateCred(map="oracle.wsm.security", key="keystore-csf-key", user="owsm",
password="MyPassword", desc="Keystore key")
updateCred(map="oracle.wsm.security", key="enc-csf-key",
user="external_webcenter_custom", password="MyPassword", desc="Encryption key")
updateCred(map="oracle.wsm.security", key="sign-csf-key",
user="external_webcenter_custom", password="MyPassword", desc="Signing key")
```

Configure the Discussions Server Connection

Supply the WS-Security client certificate information within the discussions server connection that is configured for WebCenter Portal or your Portal Framework application, as described in Section 12.3, "Registering Discussions Servers." Also see Section 36.3.3.4, "Configuring the Discussions Server Connection Settings" for example connection detail settings for the Edit Discussions and Announcement Connection page.

36.4 Securing WebCenter Portal for Applications Consuming WebCenter Portal Client API with WS-Security

This section describes the administrator tasks required to configure WS-Security for WebCenter Portal so that the communication between the an application exposing the WebCenter Portal API (the consumer) and WebCenter Portal (the producer) is secure, and that the identity of the user invoking the API is protected.

For information about the developer tasks for developing applications that consume the WebCenter Portal client API, see the "How to Set Up Your Portal Framework application to Use the WebCenter Portal API" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

This section includes the following subsections:

- Section 36.4.1, "Configuring a Simple Topology for Applications Consuming WebCenter Portal Client API"
- Section 36.4.2, "Configuring a Typical Topology for Applications Consuming WebCenter Portal Client API"

- Section 36.4.3, "Configuring a Complex Topology for Applications Consuming WebCenter Portal Client API"

36.4.1 Configuring a Simple Topology for Applications Consuming WebCenter Portal Client API

If your client application is part of the same domain as WebCenter Portal, you only need to specify the following for the `GroupSpaceWSContext()`:

```
GroupSpaceWSContext context = new GroupSpaceWSContext();
context.setRecipientKeyAlias("orakey");
```

If your client application is JDeveloper and you have access to the WebCenter Portal server's configured keystore, copy the same keystore to JDeveloper's `DefaultDomain/config/fmwconfig/dir` and configure the JDeveloper domain to use this keystore. The steps here would be exactly same as those in Section 36.1.2.2, "Configuring the Keystore with WLST," and you would then also need to specify the following on your client stub:

```
GroupSpaceWSContext context = new GroupSpaceWSContext();
context.setRecipientKeyAlias("orakey");
```

36.4.2 Configuring a Typical Topology for Applications Consuming WebCenter Portal Client API

If your client application is part of the same domain as WebCenter Portal, you only need to specify the following for the `GroupSpaceWSContext()`:

```
GroupSpaceWSContext context = new GroupSpaceWSContext();
context.setRecipientKeyAlias("webcenter");
```

If your client application is JDeveloper and you have access to the WebCenter Portal server's configured keystore, copy the same keystore to JDeveloper's `DefaultDomain/config/fmwconfig/dir` and configure the JDeveloper domain to use this keystore. The steps here would be exactly same as those in Section 36.2.2.2, "Configuring the Keystore Using WLST," and you would then also need to specify the following on your client stub:

```
GroupSpaceWSContext context = new GroupSpaceWSContext();
context.setRecipientKeyAlias("webcenter");
```

36.4.3 Configuring a Complex Topology for Applications Consuming WebCenter Portal Client API

If your client application is part of the same domain as WebCenter Portal, you only need to specify the following for the `GroupSpaceWSContext()`:

```
GroupSpaceWSContext context = new GroupSpaceWSContext();
context.setRecipientKeyAlias("webcenter");
```

If your client application is JDeveloper, copy the same keystore to JDeveloper's `DefaultDomain/config/fmwconfig/dir` and configure the JDeveloper domain to use this keystore. The steps here would be exactly same as those in Section 36.3.2.2, "Configuring the Keystore Using WLST," and you would then also need to specify the following on your client stub:

```
GroupSpaceWSContext context = new GroupSpaceWSContext();
context.setRecipientKeyAlias("webcenter");
```

37

Configuring Security for Portlet Producers

This chapter describes how to configure your WebCenter Portal or Portal Framework application to handle security for WSRP and JPDK portlet producers.

This chapter includes the following sections:

- Section 37.1, "Securing a WSRP Producer"
- Section 37.2, "Securing a PDK-Java Producer"

> **Permissions:** To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console. Users with the `Monitor` or `Operator` roles can view security information but cannot make changes.
>
> See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

37.1 Securing a WSRP Producer

The following sections describe how to secure access to JSR-168 standards-based WSRP portlets from WebCenter Portal and Portal Framework applications:

- Section 37.1.1, "Deploying the Producer"
- Section 37.1.2, "Attaching a Policy to the Producer Endpoint"
- Section 37.1.3, "Setting Up the Keystores"

For a conceptual overview of securing WSRP producers, see the "Securing Identity Propagation Through WSRP Producers with WS-Security" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

37.1.1 Deploying the Producer

Before you configure the producer for WS-Security, you must first deploy your standards-compliant portlet producer to an Oracle WebLogic managed server by performing the steps described in Section 21.11, "Deploying Portlet Producer Applications."

37.1.2 Attaching a Policy to the Producer Endpoint

This section describes how to attach a security policy to a WSRP producer endpoint. The following policies are supported for WSRP producers:

- Username token with password

  `wss10_username_token_with_message_protection_service_policy`

  This policy enforces message-level protection (message integrity and confidentiality) and authentication for inbound SOAP requests in accordance with the WS-Security 1.0 standard. It uses WS-Security's Basic 128 suite of asymmetric key technologies (specifically, RSA key mechanism for message confidentiality, SHA-1 hashing algorithm for message integrity, and AES-128 bit encryption). The keystore is configured through the security configuration. Authentication is enforced using credentials in the WS-Security UsernameToken SOAP header. The user's Subject is established against the currently configured identity store.

- Username token without password

  `wss10_username_id_propagation_with_msg_protection_service_policy`

  This policy enforces message level protection (message integrity and confidentiality) and identity propagation for inbound SOAP requests using mechanisms described by the WS-Security 1.0 standard. Message protection is provided using WS-Security's Basic 128 suite of asymmetric key technologies (specifically, RSA key mechanisms for confidentiality, SHA-1 hashing algorithm for integrity, and AES-128 bit encryption). Identity is set using the user name provided by the UsernameToken WS-Security SOAP header. The Subject is established against the currently configured identity store.

- SAML token

  There are four SAML token policies:

  – WSS 1.0 SAML token Policy:

    `wss10_saml_token_service_policy`

    This policy authenticates users using credentials provided in SAML tokens in the WS-Security SOAP header. The credentials in the SAML token are authenticated against a SAML login module. This policy can be applied to any SOAP-based endpoint.

  –

  – WSS 1.0 SAML token with message integrity:

    `wss10_saml_token_with_message_integrity_service_policy`

    This policy provides message-level integrity protection and SAML-based authentication for inbound SOAP requests in accordance with the WS-Security 1.0 standard. It uses WS-Security's Basic 128 suite of asymmetric key technologies, specifically SHA-1 hashing algorithm for message integrity.

  – WSS 1.0 SAML token with message protection:

    `wss10_saml_token_with_message_protection_service_policy`

    This policy enforces message-level protection and SAML-based authentication for inbound SOAP requests in accordance with the WS-Security 1.0 standard. It uses WS-Security's Basic 128 suite of asymmetric key technologies, specifically RSA key mechanisms for message confidentiality, SHA-1 hashing algorithm for message integrity, and AES-128 bit encryption.

  – WSS 1.1 SAML token with message protection:

    `wss11_saml_token_with_message_protection_service_policy`

This policy enforces message-level protection (that is, message integrity and message confidentiality) and SAML-based authentication for inbound SOAP requests in accordance with the WS-Security 1.1 standard. Messages are protected using WS-Security's Basic 128 suite of symmetric key technologies, specifically RSA key mechanisms for message confidentiality, SHA-1 hashing algorithm for message integrity, and AES-128 bit encryption. The keystore is configured through the security configuration. It extracts the SAML token from the WS-Security binary security token, and uses those credentials to validate users against the configured identity store. This policy can be attached to any SOAP-based endpoint.

The keystore is configured through the security configuration. It extracts the SAML token from the WS-Security binary security token, and uses those credentials to validate users against the configured identity store.

To attach a policy to a producer endpoint

1. Open Fusion Middleware Control and log into the target domain.

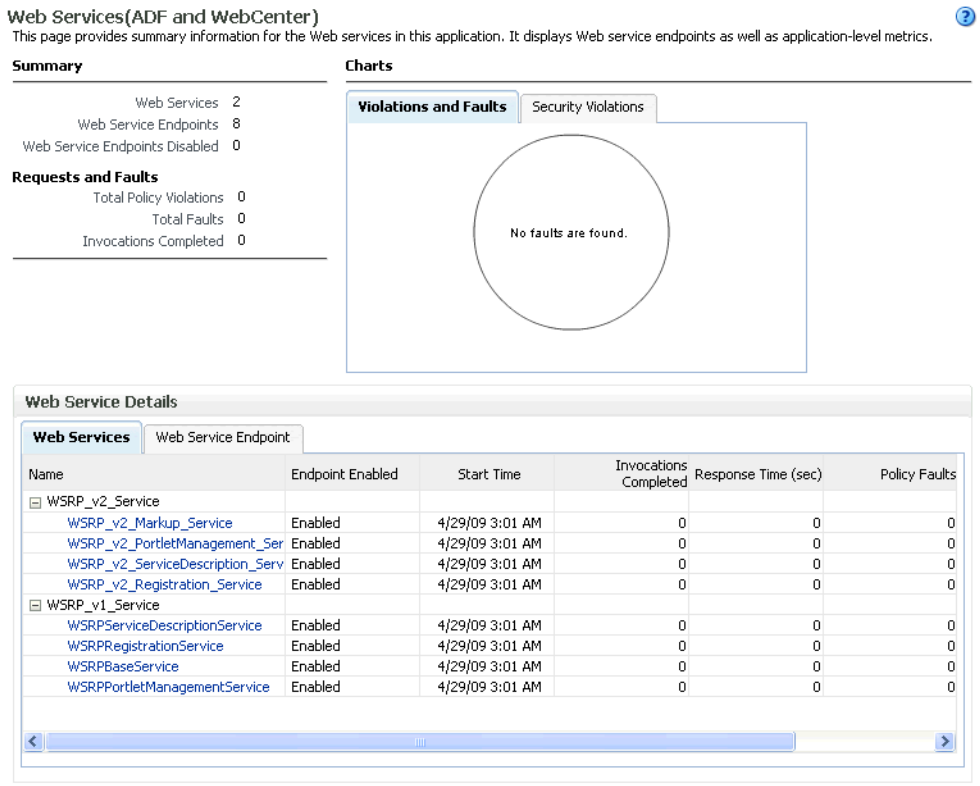
   For information on logging into Fusion Middleware Control, see Section 6, "Starting Enterprise Manager Fusion Middleware Control."

2. In the Navigation pane, expand the Application Deployments node, and click the producer to attach a policy to.

3. From the Application Deployment menu, select **Web Services**.

   The Web Services Summary page for the producer displays (see Figure 37–1).

Figure 37–1 Web Services Summary Page

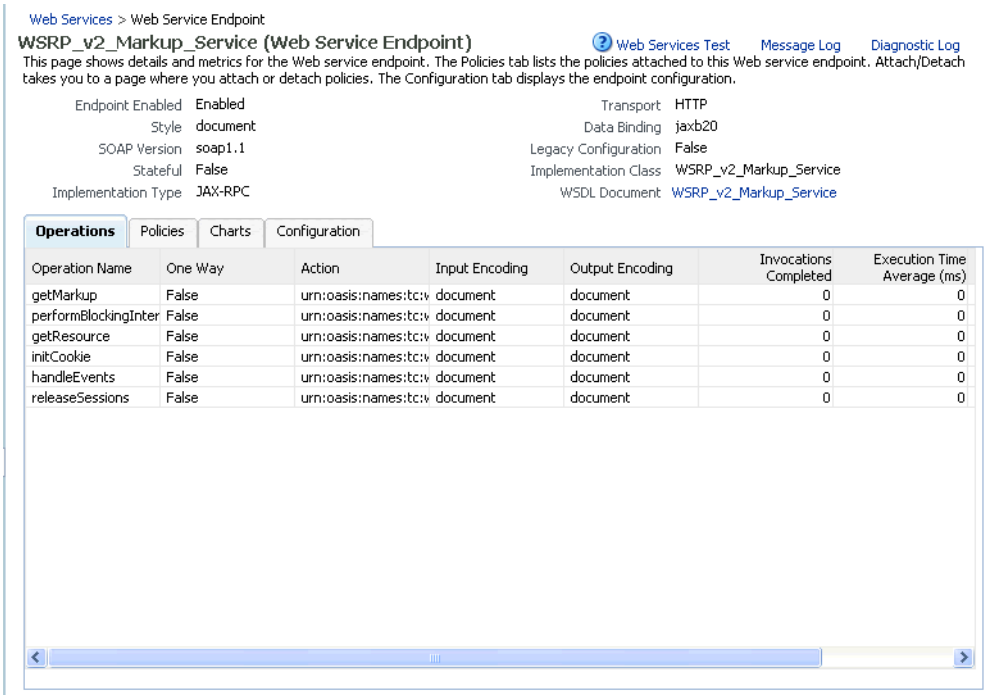


4. Open the Web Service Endpoint tab and click the endpoint to which to attach a policy.

Note: Only the markup service ports should be secured (`WSRP_V2_Markup_Service` and `WSRP_V1_Markup_Service`).

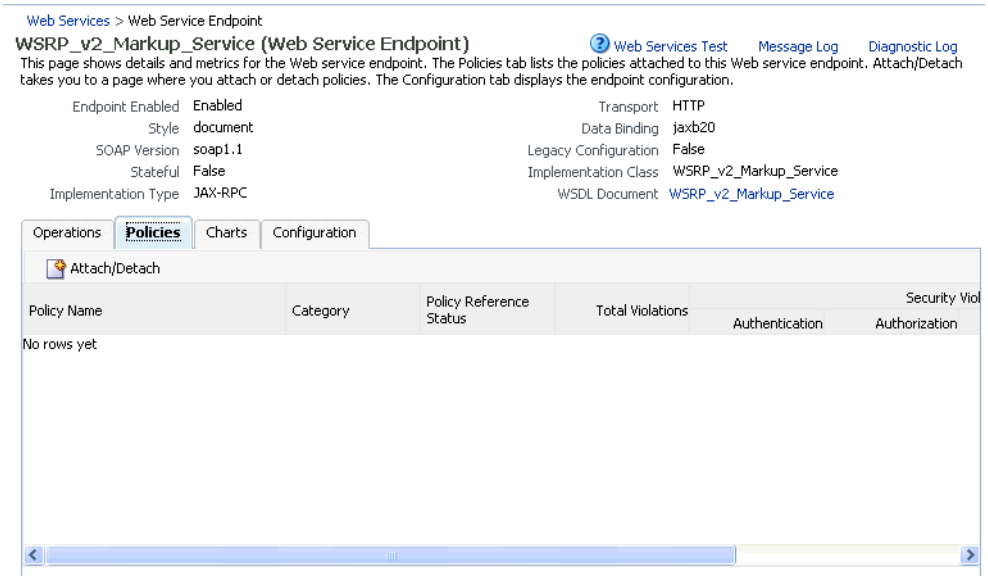
The Web Service Endpoints page for the producer displays (see Figure 37–2).

Figure 37–2 Web Service Endpoints Page



5. Open the Policies tab to display the currently attached policies for the producer (see Figure 37–3).

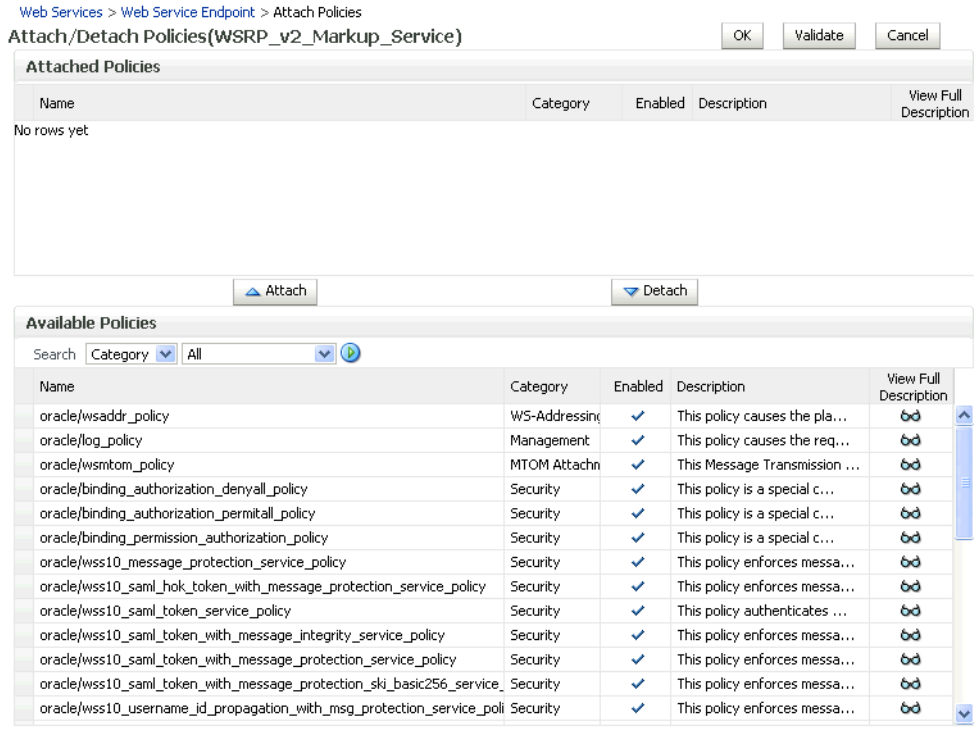
Figure 37–3 Web Services Endpoint Policies Page



6. Click **Attach/Detach** to add or remove a policy.

   The Attach/Detach Policies page is shown listing the available policies and their descriptions (see Figure 37–4).

Figure 37–4 Attach/Detach Policies Page

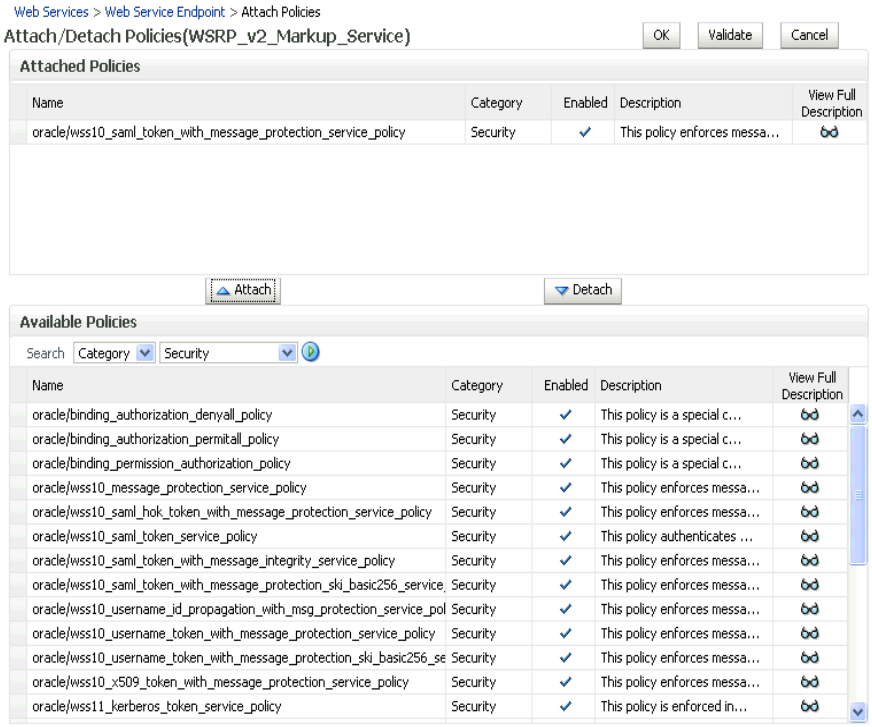


7. Under Available Policies, select `Category` and `Security` as the policy category to search, and click the Search icon to list the security policies.

8. Select the policies to attach and click **Attach**. Use the **Ctrl** key to select multiple policies.

   The policies appear in the list under Attached Policies (see Figure 37–5).

Figure 37–5 Attach Detach Policy Page with Policy Attached



9. When finished adding polices to attach to the producer endpoint, click **OK**.

37.1.3 Setting Up the Keystores

The steps to create and configure keystores for a WSRP producer depend on the topology of your WebCenter Portal or Portal Framework application environment, and are covered in the following sections:

- Section 36.1, "Configuring WS-Security for a Simple Topology"
- Section 36.2, "Configuring WS-Security for a Typical Topology"
- Section 36.3, "Configuring WS-Security for a Complex Topology"

Please refer to these sections for more complete instructions for setting up the keystores, and other WS-Security aspects of configuring WSRP producers.

37.2 Securing a PDK-Java Producer

A shared key can be defined for message integrity protection and should be used with SSL. The steps to store a shared key as a password credential are:

- Define a shared key as a password credential in the credential store of the administration server instance. This can be done using either Fusion Middleware Control or WLST.
- Restart the web producer and access the test page. Confirm that the shared key has been picked up correctly by checking the application logs.

Note: Using a shared key provides only message integrity protection. For complete message protection SSL is required. For more information on securing PDK-Java portlets using SSL, see Section 35.5, "Securing the WebCenter Portal Connection to Portlet Producers with SSL."

37.2.1 Defining a Shared Key as a Password Credential

You can define a shared key as a password credential in the credential store of the administration server instance using either Fusion Middleware Control or WLST commands, as described in the following subsections:

- Section 37.2.1.1, "Defining a Shared Key Using Fusion Middleware Control"
- Section 37.2.1.2, "Defining a Shared Key Using WLST"

37.2.1.1 Defining a Shared Key Using Fusion Middleware Control

To define a shared key using Fusion Middleware Control:

1. Log into Fusion Middleware Control.

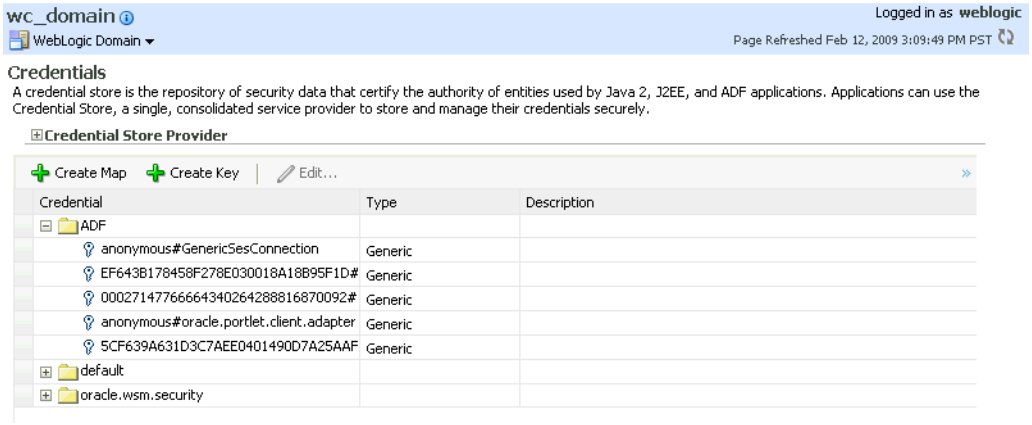
   For information on logging into Fusion Middleware Control, see Section 6, "Starting Enterprise Manager Fusion Middleware Control."

2. In the Navigation pane, expand the WebLogic Domain node and click the target domain (for example, `wc_domain`).

3. From the WebLogic Domain menu, select **Security > Credentials**.

   The Credentials pane displays (see Figure 37–6).

Figure 37–6 Credentials Pane



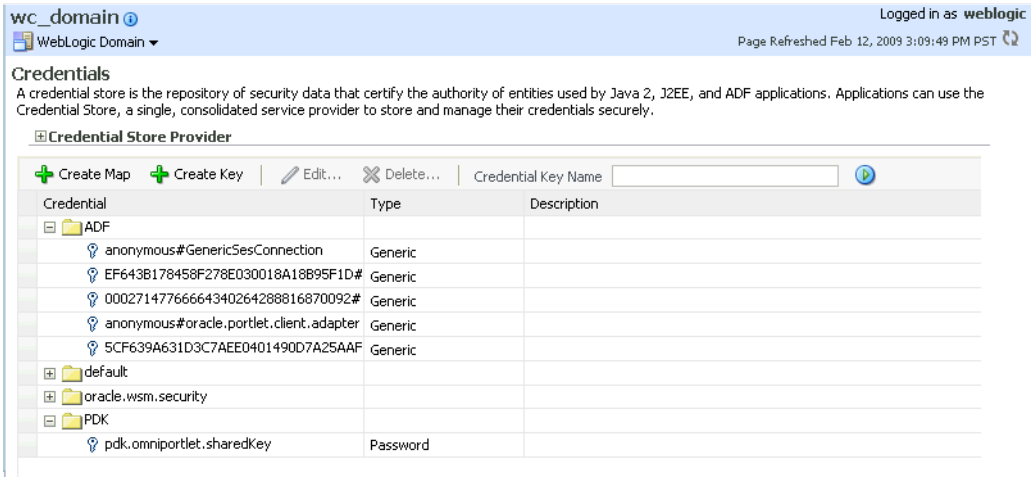
4. Click **Create Map** and enter `PDK` as the **Map Name** and click **OK**.

5. Click **Create Key** and select the map (`PDK`) you just created.

6. Enter a **User Name** (this value is not used so it could be anything), a **Key** in the form `pdk.<service_id>.sharedKey` (where `<service_id>` is the name of the producer), and a 10 to 20 hexadecimal digit **Password** and click **OK**.

   The new key is displayed in the Credential pane (see Figure 37–7).

Figure 37–7 Credentials Pane with New Shared Key



37.2.1.2 Defining a Shared Key Using WLST

You can also define a shared key using WLST as described in the following steps:

1. Start WLST as shown in Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands," and connect to the Administration Server instance for the target domain.
2. Connect to the Administration Server for the target domain with the following command:

   ```
   connect('user_name','password, 'host_id:port')
   ```

   Where:

   - *user_name* is the name of the user account with which to access the Administration Server (for example, weblogic)
   - *password* is the password with which to access the Administration Server
   - *host_id* is the host ID of the Administration Server
   - *port* is the port number of the Administration Server (for example, `7001`).
3. Add a shared key credential for a producer to the credential store using the WLST `createCred` command:

   ```
   createCred(map='PDK', key='pdk.service_id.sharedKey.user_name',
   user='user_name', password='password')
   ```

   Where:

   - *service_id* is the name of the producer to create the key for (for example, `omniPortlet`)
   - *user_name* is the name of the user. This value is not used so it could be anything.
   - *password* is a 10 to 20 hexadecimal digit value.

   For example:

   ```
   createCred(map='PDK', key='pdk.omniPortlet.sharedKey', user='sharedKey',
   password='1234567890abc')
   ```

Note: After creating a credential, you can use the WLST `updateCred` command with the same parameters as above to update it.

4. Restart the producer.

   Web producers pick up properties the first time they handle a request (for example, a browser test page request or when they are first registered), so producers should be restarted once a shared key credential has been set up.

37.2.1.3 Registering a PDK-Java Producer with a Shared Key

Registering a PDK-Java producer is described in Section 21.4, "Registering Oracle PDK-Java Producers." When you register a PDK-Java producer with a shared key, you must be sure to also do the following:

- Select the **Enable producer session** option when registering the producer.
- In the **Add Portlet Producer Connection** section, enter the password used when creating the credential map as the **Shared Key**.

38

Managing Impersonation

This chapter describes how to manage and configure WebCenter Portal Impersonation, which lets designated users impersonate other portal users and perform operations as those users. For instructions on how to initiate an impersonation session (by the impersonator) and how to allow an Impersonation session (by the impersonatee), see the "Using WebCenter Portal Impersonation" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*. For information about impersonation ELs and APIs, see the "ELs Related to Impersonation" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

This chapter includes the following sections:

> **Permissions:** To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console. Users with the `Monitor` or `Operator` roles can view security information but cannot make changes.
>
> See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

38.1 Introduction to WebCenter Portal Impersonation

This section includes the following sub-sections:

38.1.1 About WebCenter Portal Impersonation

WebCenter Portal Impersonation lets a WebCenter Portal administrator or system administrator assign impersonation rights to a group of users ("impersonators"), such as support representatives or application administrators, so that they can perform operations as other users ("impersonatees"). Note that this is subject to the impersonatee granting the impersonator additional rights to impersonate them. This may be useful in the following instances:

- A customer support representative may want to perform actions as another user in order to understand the issues being faced by that user.
- An administrator may want to perform operations on behalf of a user.
- A company executive may need to delegate someone to act on his or her behalf while away.

38.1.2 Best Practices for Using WebCenter Portal Impersonation

All applications participating in Oracle Access Manager (OAM) from an impersonatee's system will also be accessible to an impersonator. The only exception to this is that an impersonator will not be able to access the Impersonation task flow and grant or modify impersonation rights. Consequently, administrators should exercise extreme caution when granting impersonation rights because of what an impersonator could potentially access. Impersonators should be a very limited group.

Audit logging should be turned on for impersonation and the administrator should monitor the audit logs periodically to review the impersonation activities. For more information about audit logging, see Section 38.8, "Managing Audit Logs for WebCenter Portal Impersonation."

To initiate an impersonation session the impersonatee and impersonator should agree on an appropriate time slot for the impersonation session. The impersonatee should then grant impersonation rights for that time slot only. The impersonatee should revoke impersonation rights immediately after the impersonator is done.

Note that an impersonation session will end if the impersonator logs out. An impersonation session will also end when the specified impersonation time duration end point is reached. For example, if a user grants impersonation rights to an impersonator between 1:00 and 2:00 in the afternoon, although the impersonator can start an impersonation session anytime between 1:00 and 2:00, the session will end at 2:00.

Also note that if a user revokes an impersonation grant explicitly while the impersonator is in the middle of an impersonation session, the revoke will not affect any existing impersonation session for that user. It will only take effect the next time the impersonator tries to impersonate the user. The user will then not appear in the list of available impersonatees.

38.2 Preparing WebCenter Portal for Impersonation

WebCenter Portal impersonation relies on OAM 11.1.2.0. Before you can enable impersonation for a WebCenter Portal instance you must first install and configure OAM 11g (Oracle's single sign-on solution), and then turn on impersonation in OAM. For information about installing and configuring OAM 11*g*, see Section 33.2, "Configuring Oracle Access Manager (OAM)."

This section includes the following subsections:

- Section 38.2.1, "WebCenter Portal Impersonation Requirements"

- Section 38.2.2, "Turning on Impersonation in OAM"
- Section 38.2.3, "Adding Impersonation Attributes to the Identity Store"

38.2.1 WebCenter Portal Impersonation Requirements

To prepare WebCenter Portal for impersonation you must first install and configure OAM 11.1.2.0 and then turn on impersonation in OAM. You will also need to add impersonation attributes for each participating user.

> **Note:** WebCenter Portal Impersonation requires that OAM 11.1.2.0 be installed and configured as the single sign-on solution, and that OID 11.1.2.0 is installed and configured as the identity store.

- Install and configure OAM 11.1.2.0 with either the 10g or 11g WebGate (see Section 33.2, "Configuring Oracle Access Manager (OAM)")
- Turn on impersonation
- Add impersonation attributes to each participating user in the identity store
- Configure each participating WebCenter Portal instance for impersonation
- Configure the people who have impersonation rights by adding them to a WebCenter application role

38.2.2 Turning on Impersonation in OAM

After installing and configuring OAM 11.1.2.0 (with either the 10g or 11g WebGate) as described in Section 33.2, "Configuring Oracle Access Manager (OAM)," and then enable impersonation by editing the `oam-config.xml` file as shown below.

To enable impersonation, do the following in your OAM installation:

1. Locate your OAM installation and back up the `oam-config.xml` file (`DOMAIN_HOME/config/fmwconfig/oam-config.xml`).
2. Open the `oam-config.xml` file for editing and set `ImpersonationConfig` to `true`:

```
<Setting Name="ImpersonationConfig" Type="htf:map">
     <Setting Name="EnableImpersonation" Type="xsd:boolean">true</Setting>
</Setting>
```

3. Save `oam-config.xml`.
4. Restart OAM and then the OAM Managed Server.

38.2.3 Adding Impersonation Attributes to the Identity Store

For users to be available as impersonators or impersonatees they need to have the following attributes available for storing the impersonation grants in OID:

- `orclImpersonationGrantee`
- `orclImpersonationGranter`

These attributes are a part of the `orclIDXPerson` object class that is available by default in OID. This object class must be added to the list of object classes for each user's user record that you want to participate as an impersonator or impersonatee.

You can do this either by adding the object class to individual users, or as a bulk update for multiple users as described in the following subsections:

- Section 38.2.3.1, "Adding Impersonation Attributes for Individual Users"
- Section 38.2.3.2, "Adding Impersonation Attributes for Multiple Users"

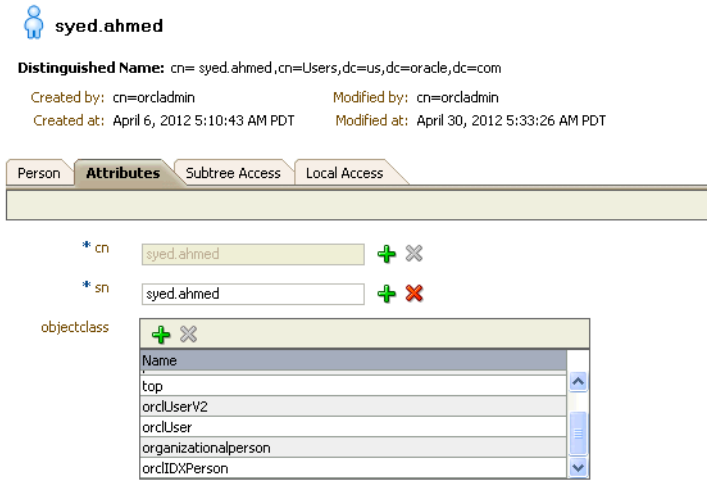
38.2.3.1 Adding Impersonation Attributes for Individual Users

Follow the steps below to add the attributes for storing the impersonation grants in OID for individual users:

To add the object class to individual users:

1. Log in to ODSM (typically `http://host:port/odsm`).
2. Connect to the directory that is configured for OAM and WebCenter.
3. For each participating user:
   a. Locate the user you want to change by drilling down in the DataBrowser, or by using the DataBrowser's search field.
   b. Open the Attributes screen and add the `orclIDXPerson` object class to the list of existing object classes as shown in Figure 38–1.

Figure 38–1 ODSM Attributes Tab



   c. Click **Apply**.

38.2.3.2 Adding Impersonation Attributes for Multiple Users

You can add the attributes available for storing the impersonation grants in OID as a bulk update using the `bulkmodify` tool. Note that to use this tool you need to be able to access the machine where OID is installed, have system administrator rights, and need to know the OID database password.

To add the attributes for storing impersonation grants in OID for multiple users:

1. Stop OID.
2. Go to `$ORACLE_HOME/ldap/bin` and run the `bulkmodify` tool.

   Specify `basedn` as the DN under which all users you wish to add the object class reside. The connect string is the OID DB connect string, which is typically `OIDDB` (determined from `$ORACLE_INSTANCE/config/tnsnames.ora`). Provide the

DB password when prompted. The following shows a sample run of the command:

```
bulkmodify connect="OIDDB" basedn="cn=Users,dc=us,dc=oracle,dc=com"
attribute="objectclass" value="orclIDXPerson" add=true
This tool can only be executed if you know database user password for OID
Enter OID Password ::

-------------------------------------------------------------
Modifying entries under "cn=users,dc=us,dc=oracle,dc=com" ...
-------------------------------------------------------------

-------------------------------------------------------------
Total 72 Entries are modified.
```

3. Restart OID.

   All users under the specified DN should now have the `orclIDXPerson` object class configured. For more information about the `bulkmodify` tool, see the "bulkmodify" section in *Oracle Fusion Middleware Reference for Oracle Identity Management*.

38.3 Configuring WebCenter Portal for Impersonation

After installing and configuring OAM and enabling Impersonation in OAM, you need to configure the OAM Impersonation trigger end points in your WebCenter Portal instance as shown below:

1. Using WLST, connect as administrator to the Weblogic Administration Server and run the following command replacing *`oamhost`* and *`oamserverport`* with the corresponding host ID and port for OAM:

   ```
   addOAMSSOProvider(loginuri="/${app.context}/adfAuthentication",
   logouturi="/oamsso/logout.html",
   beginimpuri="http://oamhost:oamserverport/oam/server/impersonate/start",
   endimpuri="http://oamhost:oamserverport/oam/server/impersonate/end")
   ```

2. Restart all servers in the WebCenter Portal domain, including the Admin Server.

3. You may also need to account for any time difference between your WebCenter Portal server and OAM. Although Impersonation start and end times are accepted in WebCenter Portal, they are enforced by OAM so the time settings must be consistent. To account for time differences:

   a. Log into WebCenter Portal as an administrator.

   b. Select **Administration > Attributes**.

      The Attributes page displays.

      Tip: You can also access the Attributes page directly by opening the page in your browser:

      `http://host:port/webcenter/faces/oracle/webcenter/webcenterapp/view/pages/admin/WebCenterAdmin-CustomAttributes.jspx`

      where `host` and `port` are the host and port IDs of the `WC_Spaces` server.

   c. Specify the Impersonation time Delta in seconds using a **+** sign if the WebCenter Portal server is behind the OAM server, or a **-** sign if it is ahead. For example:

```
oracle.webcenter.security.impersonation.timedelta = -480
```

would indicate that there is a time difference of eight minutes between OAM and WebCenter Portal with the WebCenter Portal server being ahead.

Tip: You can also add the setting to the `$domain.home/bin/setDomainEnv.sh` file:

```
EXTRA_JAVA_PROPERTIES="-Doracle.webcenter.spaces.osso=true
-Doracle.webcenter.security.impersonation.timedelta=-480"
export EXTRA_JAVA_PROPERTIES
```

d. Restart the WebCenter Portal managed server (`WC_Spaces`).

38.4 Configuring Impersonators

After configuring OAM and WebCenter Portal, you must configure the users to whom you want to grant impersonation privileges by adding those users or groups to the `webcenter#-#impersonators` role. Out-of-the-box, no users are granted this role. Only users belonging to this role either by direct membership or through an enterprise role membership are eligible to impersonate users in a WebCenter Portal instance.

Caution: Use caution when granting rights to users that would allow them to impersonate other users. Only users that have a business need for this feature should be granted impersonation rights. For information about best practices, see Section 38.1.2, "Best Practices for Using WebCenter Portal Impersonation."

Use the `grantAppRole` WLST command to grant the `webcenter#-#impersonators` role to one or more enterprise roles or users. For example:

- To grant the impersonators role to an enterprise role called `SupportRepresentatives`:

```
grantAppRole(appStripe="webcenter", appRoleName="webcenter#-#impersonators",
principalClass="weblogic.security.principal.WLSGroupImpl",
principalName="SupportRepresentatives")
```

- To grant the impersonators role to a user named `weblogic`:

```
grantAppRole(appStripe="webcenter", appRoleName="webcenter#-#impersonators",
principalClass="weblogic.security.principal.WLSUserImpl",
principalName="weblogic")
```

Use the `revokeAppRole` WLST to revoke impersonator permission from an enterprise role or user. For example:

- To revoke the impersonators role from an enterprise role called `SupportRepresentatives`:

```
revokeAppRole(appStripe="webcenter", appRoleName="webcenter#-#impersonators",
principalClass="weblogic.security.principal.WLSGroupImpl",
principalName="SupportRepresentatives")
```

- To revoke the impersonators role from a user named `weblogic`:

```
revokeAppRole(appStripe="webcenter", appRoleName="webcenter#-#impersonators",
principalClass="weblogic.security.principal.WLSUserImpl",
```

```
principalName="weblogic")
```

Note: Changes to role assignments are available immediately. You do not need to restart the managed server.

38.5 Disabling Impersonation

WebCenter Portal Impersonation is disabled by default, so unless you have already enabled impersonation there is nothing that needs to be done to turn it off. However, if you have enabled it and now want to disable it, follow the steps below to turn it off in WebCenter Portal and OAM.

Note that turning off impersonation in WebCenter Portal only disables it for that particular instance. Any other WebCenter Portal instances for which impersonation was enabled will not be affected until you turn off impersonation in OAM.

To disable impersonation for WebCenter Portal:

1. Log into Fusion Middleware Control as an administrator.
2. Go to **WebCenter Domain > Security > Security Provider Configuration.**
3. Navigate to the Properties section and click **Configure**.
4. Under **PropertySets**, locate the property set that defines the impersonation start and stop URIs (typically "`props.auth.uri.0`").
5. Delete the properties `imp.begin.url` and `imp.end.url`.
6. Restart all servers in the WebCenter Portal domain, including the Admin server.

   Note that until you disable impersonation in OAM, impersonation in other WebCenter Portal domains will continue to be enabled.

To disable impersonation in OAM and turn off impersonation altogether:

1. Back up the `DOMAIN_HOME/config/fmwconfig/oam-config.xml` file.
2. Open the `oam-config.xml` file for editing.
3. Set `ImpersonationConfig` to false as shown below:

   ```
   <Setting Name="ImpersonationConfig"Type="htf:map"> <Setting
   Name="EnableImpersonation"Type="xsd:boolean">false</Setting> </Setting>
   ```

4. Save `oam-config.xml`.
5. Restart OAM and all of its components.

38.6 Turning off the Session Indicator

The session indicator is an overlay that appears on the impersonator's screen by default during an impersonation session. Although the overlay provides a visual clue that the impersonation session is active, and also provides a quick way to stop the session by clicking **Stop Impersonation**, it may obstruct a view of part of the user's (impersonatee's) screen as show in Figure 38–2.

Note: When the impersonation session notification toolbar is turned off, users must use the Impersonation page to stop an impersonation session since the **Stop Impersonation** button will no longer be visible.

Figure 38–2 Impersonation Session - Session Indicator Overlay

You can turn off the session indicator overlay as shown below:

To turn off the session indicator:

1. Log into WebCenter Portal as an administrator.
2. Select **Administration > Attributes**.

   The Attributes page displays.

   Tip: You can also access the Attributes page directly by opening the page in your browser:

   `http://host:port/webcenter/faces/oracle/webcenter/webcenterapp/view/pages/admin/WebCenterAdmin-CustomAttributes.jspx`

   where *host* and *port* are the host and port IDs of the `WC_Spaces` server.

3. Set the notification property to `false` as shown below:

   ```
   oracle.webcenter.security.impersonation.notification=false
   ```

   Note that impersonators will now need to end impersonation sessions using the Impersonation Preferences screen. For more information about using the Impersonation Preferences screen, see "the Using WebCenter Portal Impersonation" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.
4. Restart the `WC_Spaces` managed server for the change to take effect.

38.7 Overriding the Impersonation Hotkey

The default `Ctrl-Shift-I` hotkey sequence used by the impersonator to view the list of impersonatees can be overridden, if needed.

To change the hotkey sequence:

1. Log into WebCenter Portal as an administrator.
2. Select **Administration > Attributes**.

   The Attributes page displays.

   Tip: You can also access the Attributes page directly by opening the page in your browser:

   `http://host:port/webcenter/faces/oracle/webcenter/webcenterapp/view/pages/admin/WebCenterAdmin-CustomAttributes.jspx`

   where *host* and *port* are the host and port IDs of the `WC_Spaces` server.

3. Set the new hotkey sequence as shown below:

   ```
   oracle.webcenter.security.impersonation.key=new key
   ```

where *new key* is a single character to be appended to `Ctrl-Shift`. Note that you can only override the default "I" with another single character. The `Ctrl-Shift` sequence is predefined and will always precede the key. Be sure to check that the overridden character is not already used by other components, tools or plug-ins. For example, `Ctrl-Shift-M` is used by menus, and `Ctrl-Shift-K` and `Ctrl-Shift-J` are sometimes used by browser plug-ins such as developer tools and the error console.

4. Restart the `WC_Spaces` server for the change to take effect.

38.8 Managing Audit Logs for WebCenter Portal Impersonation

WebCenter Portal Impersonation, when enabled, activates logging for Impersonation-related events as part of the Fusion Middleware Audit Service. Audit log events are stored in a file (the Audit Bus-stop) by default, but can also be uploaded to a database for persistency.

Note: If you enable WebCenter Portal Impersonation, it is highly recommended that you also enable audit logging. When Impersonation is enabled, audit logging tracks the impersonator, impersonatee, and the context surrounding each impersonation event.

The Audit Bus-stop file has a limited capacity so storing log information in a database where events can be queried long after their occurrence is also recommended.

Impersonation audit logging provides the following key benefits:

- Events that alter the security settings of Portal, Portal Server, and major Portal Server artifacts are traceable
- Auditable events contain all relevant event payload to help define the impersonator, impersonatee and the context surrounding an event
- Definable logging levels
- Events logged are available in perpetuity when uploaded to a database
- Reports on audit events are available through the Audit Service

For more information about managing audit logging for WebCenter Portal, see Chapter 29, "Managing Oracle WebCenter Portal Audit Logs." For information about configuring the Audit Service to use a database, see the "Configuring and Managing Auditing" section in *Oracle Fusion Middleware Application Security Guide*.

Part VIII

Lifecycle: WebCenter Portal

This part of *Oracle Fusion Middleware Administering Oracle WebCenter Portal* presents WebCenter Portal lifecycle operations.

Part VIII contains the following chapters:

- Section 39, "Understanding WebCenter Portal Life Cycle"
- Section 40, "Deploying Portals, Templates, Assets, and Extensions"
- Section 41, "Managing WebCenter Portal Backup, Recovery, and Cloning"

39

Understanding WebCenter Portal Life Cycle

This chapter discusses tasks, tools, and techniques for managing WebCenter Portal throughout its life cycle.

This chapter includes the following topics:

> **Note:** Portal Framework applications have a different life cycle. For more information, see the "Understanding the WebCenter Portal Framework Application Life Cycle" chapter in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

Permissions: To perform the tasks in this chapter, you must be granted the following roles:

- **WebLogic Server:** `Admin` role granted through the Oracle WebLogic Server Administration Console.
- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration.

See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

39.1 What is the WebCenter Portal Life Cycle?

The portal life cycle describes the process of creating a portal using Portal Builder through deployment to a production instance. Many actors participate in the life cycle including software developers, content modelers, content contributors, IT administrators, portal site administrators. The phases of the life cycle typically include development, testing, staging, and production. Each phase requires certain tasks to be performed. Some tasks are performed only once, like setting up a content repository. Others are performed more frequently, like creating backups, and performing nightly builds. The phases of the portal life cycle are described in Table 39–1.

Table 39–1 WebCenter Portal Life Cycle Phases

Life Cycle Phase	Primary Actors/Roles	Description
Development	■ Portal Developers ■ Web Developers ■ Content Modelers ■ Content Contributors ■ Application Specialists	Developers can use *Portal Builder*, WebCenter Portal's browser-based tool for developing new portals. For advanced requirements, developers can use JDeveloper to further develop and deploy portal assets and shared libraries (containing custom portal components). The development portal typically employs test data and content. Some of the features that are developed in this phase of the life cycle include: ■ portals ■ portlets ■ task flows ■ shared libraries ■ skins ■ navigation models ■ page templates ■ display templates ■ content models ■ data transfer and interportlet communication ■ initial security

Table 39–1 (Cont.) WebCenter Portal Life Cycle Phases

Life Cycle Phase	Primary Actors/Roles	Description
Testing	■ Developers ■ QA Engineers ■ System Administrators	The development portal is deployed to an independent testing environment. The test environment typically includes its own Metadata Service (MDS) and policy store that are database-based, and has a dedicated Oracle WebCenter Content instance. The testing environment may contain test data and test content that will not become part of the production portal. Portlet producers may be shared between the test and development environments. However, if the usage load is high, Oracle recommends that separate instances be created.
Staging	■ Application Specialists ■ System Administrators ■ Content Contributors	The staging environment provides a stable environment where final configuration and testing takes place before the portal is moved to production. Content contributors add content and refine the portal structure. Typically, the staging environment includes a dedicated Oracle WebCenter Content server, as well as a dedicated portlet producer server (WC_Portlet), a utilities server for analytics, activity graph, data integration (WC_Utilities), and a collaboration server for discussions and announcements (WC_Collaboration). The staging server is often maintained as a mirror of the production site.
Production	■ Application Specialists ■ System Administrators ■ Content Contributors ■ Knowledge Workers	A production portal is live and available to end users. A portal in production can be modified whilst online with tools like Portal Builder, the Assets page, and Composer. For instance, an administrator might add additional portlets to a portal or reconfigure the content of a portal. Individual users with proper authorization can also customize their view. WebCenter Portal provides a set of WLST commands for migrating portals and content to the production environment. See Section 40.7, "Moving Portals from Staging to Production." Some back-end data must be moved manually. For details, see Section 40.8, "Moving External Portal Data from Staging to Production." **Note:** Administrators can propagate *metadata changes* in staging to production providing that the two environments are kept "in sync", that is, by always making changes in stage first and then pushing the metadata changes to production using deployment or propagation. For details, see Section 40.9, "Managing Portals in Production."

39.2 What Are the Major WebCenter Portal Life Cycle Tasks?

Each phase of the life cycle requires actors (developers, administrators, content contributors, and others) to perform certain tasks. This provides an overview of the kinds of tasks that are performed during each phase of the portal life cycle.

- Section 39.2.1, "One-Time Setup Tasks"
- Section 39.2.2, "Development Environment Tasks"
- Section 39.2.3, "Stage Environment Tasks"
- Section 39.2.4, "Production Environment Tasks"

39.2.1 One-Time Setup Tasks

You must perform certain preparatory steps to set up development, test, stage, and production environments for WebCenter Portal. Table 39–2 provides a general list of these preliminary setup tasks and the environments to which they apply.

See Section 39.7, "Setting Up a Staging or Production WebCenter Portal Environment for the First Time."

Table 39–2 Typical One-Time Setup Tasks

Setup Task	Development in JDeveloper (Assets and Shared Libraries only)	Development/Test in Portal Builder	Stage	Production
Install Oracle JDeveloper and WebCenter Portal extension for JDeveloper	Yes	No	No	No
Install Oracle WebCenter Portal	No	Yes	Yes	Yes
Install Oracle WebLogic Server; create a domain and managed servers	No	Yes	Yes	Yes
Create required database schemas using RCU	No	Yes	Yes	Yes
Install and configure Oracle WebCenter Content	Yes	Yes	Yes	Yes
Install identity management components, such as Oracle Access Manager	No	Yes	Yes	Yes
Create the required Oracle Platform Security Services policies in the policy store	No	Yes	Yes	Yes
Create required user credentials in the credential store	No	Yes	Yes	Yes
Create connections to back end servers	Yes	Yes	Yes	Yes
Set up source control and nightly build scripts	Yes	No	No	No
Create deploy and configure scripts	No	Yes	Yes	Yes
Create backup scripts	No	No	Yes	Yes
Integrate/configure personalization for WebCenter Portal	Yes	Yes	Yes	Yes

39.2.2 Development Environment Tasks

Developers can use *Portal Builder*, WebCenter Portal's browser-based tool for developing new portals and portal components.

For advanced requirements, developers can use JDeveloper to further develop and deploy portal assets, shared libraries (containing custom portal components), and portlets. In a JDeveloper development environment, each developer has a local JDeveloper instance that is connected to a source control system and a shared Oracle WebCenter Content repository.

39.2.3 Stage Environment Tasks

WebCenter Portal plays a bigger role in the staging environment than in the development environment. Nonetheless, occasional updates from development to portlets, task flows, and portal assets will need to be deployed to the stage environment. To facilitate these more intermittent updates, WebCenter Portal provides a set of WLST commands, as well as comparable menu options in WebCenter Portal, that enable you to import *portal asset* updates from development to stage. If you want to update portlets and task flows on the staging environment then you redeploy them in the usual way.

For more information, see Section 39.4.1, "Setting Up a Staging Environment for WebCenter Portal" and Chapter 40, "Deploying Portals, Templates, Assets, and Extensions."

39.2.4 Production Environment Tasks

When changes are tested and approved in the stage environment they need to be pushed to the production environment. WebCenter Portal provides a set of WLST commands, as well as import/export options in WebCenter Portal, to push the entire portal server or individual portals, portal templates and assets deployed on stage to production. For more information, see Section 41.5, "Migrating Entire WebCenter Portal to Another Target." and Section 40.7, "Moving Portals from Staging to Production."

In addition, WebCenter Portal provides a WLST command that only propagates changes to portal metadata from stage to production server. For more information, see Section 40.9.3, "Propagating Portal Changes in Staging to Production."

During the export/import process you can, where required, isolate and modify connection information that is variable between environments, like the server names, ports, content management connections, and so on.

Note: When you deploy changes and updates from stage, new connections that do not exist in the stage environment are created on the target but existing connections configured on production are *never* modified.

39.3 Who Participates in the WebCenter Portal Life Cycle?

Many different people participate in the portal life cycle. In general, these people (the primary actors in Table 39–1) fall into one or more of these general roles:

- **Developer** – Uses JDeveloper to build/extend/deploy portlets, task flows, portal assets, and shared libraries for WebCenter Portal, and also manages source control.
- **Web developer** – Uses Portal Builder to build and develop portals.
- **System administrator** – Uses WLST commands to move portals between different environments. Creates backups and restores from backups. Maintains build and source control systems.

  Has full administrative privileges for WebCenter Portal. Uses WebCenter Portal administration to modify global layouts, enable tools and services, delegate security settings, and more.
- **Application specialist** – Uses the Assets page to extend the portal structure and define reusable components for the portal.
- **Content modeler** – Uses Oracle WebCenter Content's Site Studio designer to model content.
- **Content contributor** – Develops whatever content appears in or is available from the portal, including images, document files, video and audio content, and so on.
- **Knowledge worker** - End user or consumer of the production portal. Finds, creates, and updates content. Collaborates with other business users.

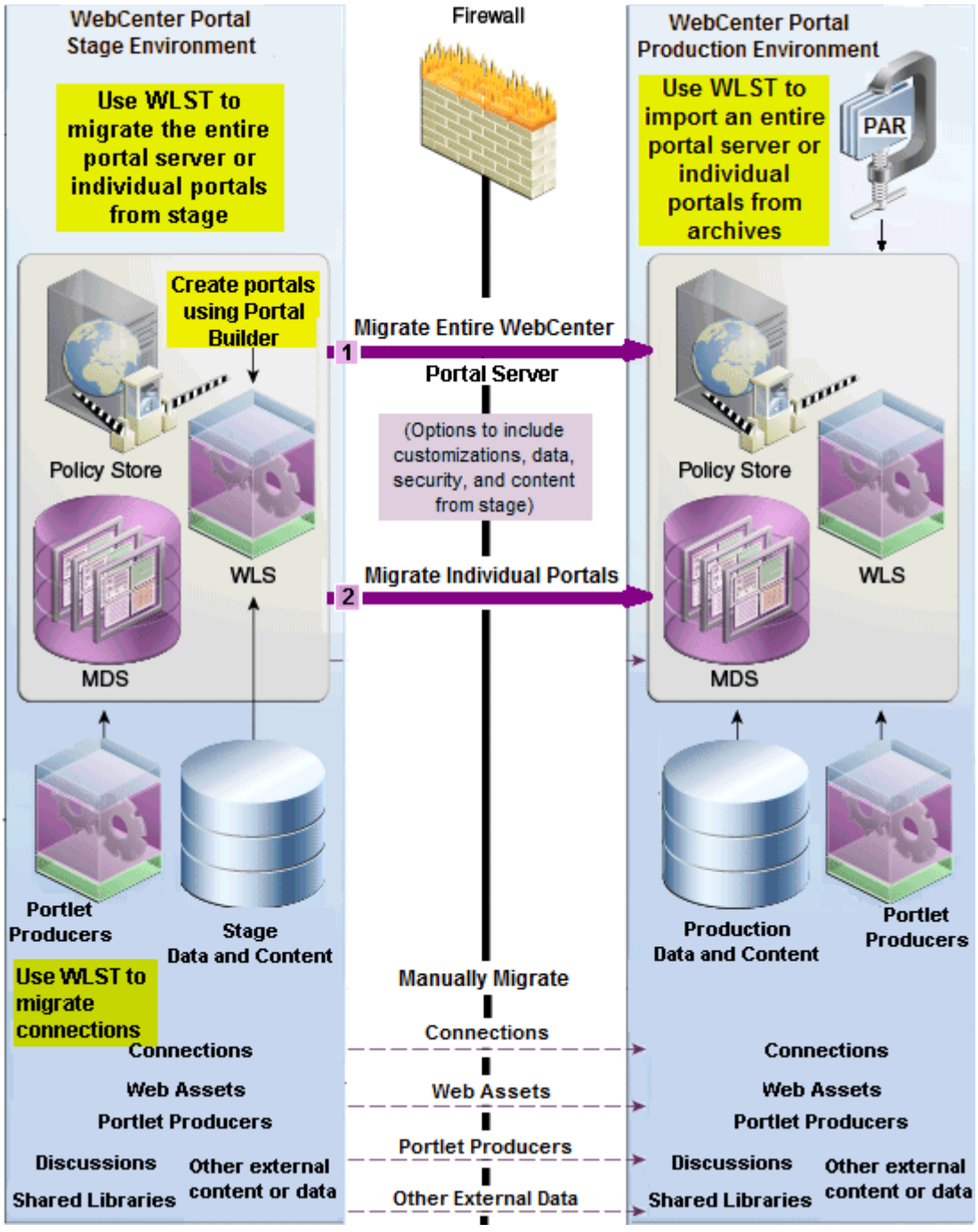
39.4 Understanding WebCenter Portal Staging and Production Environments

This section discusses the staging and production phases of portal life cycle. Figure 39–1 illustrates the general flow from staging to production environments. The figure shows that initially you migrate the entire WebCenter Portal instance from staging to the production environment.

Subsequent updates to portals and any new portals that are developed on stage can then be migrated to production as and when required. You can migrate updates on stage *directly* to production using deploy or propagate WLST commands or you can export stage updates to an file and import them on the production server through Portal Builder or using export and import WLST commands.

Note: Figure 39–1 does not depict all possible portal features.

Figure 39–1 Flow from WebCenter Portal Staging to Production Environments



39.4.1 Setting Up a Staging Environment for WebCenter Portal

The staging environment provides a stable environment where final configuration and testing takes place before the portal is moved to production. Typically, the staging environment includes a dedicated content server, discussions server, as well as dedicated portlet producer servers and utilities server (for analytics, activity graph, data integration). An external LDAP-based identity store, such as Oracle Internet Directory, must be set up for the staging environment too. For a list of typical setup tasks, see Table 39–2, " Typical One-Time Setup Tasks".

If you are setting up the staging environment for the first time, see:

- Section 39.7, "Setting Up a Staging or Production WebCenter Portal Environment for the First Time"

For information on making incremental changes to the staging environment, see:

- Section 40.7, "Moving Portals from Staging to Production"
- Section 40.6, "Moving Connections Details from Staging to Production"
- Section 40.8, "Moving External Portal Data from Staging to Production"

39.4.2 Adding Content to the WebCenter Portal Staging Environment

Content developers can add content directly to the staging server. Content workflow features in Oracle WebCenter Content can be used to manage content approvals. WebCenter Portal also provides browser tools for creating and editing portal content.

When you create or deploy a portal, WebCenter Portal creates a dedicated folder for the portal on the back-end content server. When you move a portal from stage to production you can optionally, move the portal's content folder along with the portal. See Section 40.7, "Moving Portals from Staging to Production."

39.4.3 Moving a Portal from Staging to Production

Once the staging environment is fully provisioned and tested, it can be moved to the production environment and made accessible to users.When you copy the staging environment to production for the first time, you migrate the entire stage WebCenter Portal instance to the production environment. Subsequently, and once the production environment is live, you can make incremental updates to portal metadata, content, and assets using WLST commands, automated WLST scripts, and/or replication techniques.

Typically, portal updates are performed by a system administrator. For more information, see Section 40.7, "Moving Portals from Staging to Production."

During the update process you can, where required, isolate and modify connection information that is variable between environments, like the server names, ports, content management connections, and so on.F or details, see Section 40.6, "Moving Connections Details from Staging to Production."

Note: When you deploy changes and updates from stage, new connections that do not exist in the stage environment are created on the target but existing connections configured on production are *never* modified.

New/updated portal templates and assets can be pushed to the production server by application specialists. See Section 40.2, "Deploying Portal Templates" and Section 40.3, "Deploying Assets."

39.5 Tools for Managing WebCenter Portal Life Cycle

WebCenter Portal provides a set of tools and utilities that enable you to design, build, and deploy portals, and move information between development—test—stage—production environments or WebCenter Portal installations:

- **Oracle JDeveloper and Oracle ADF** – Oracle JDeveloper and Oracle ADF provide the tools and framework to build and deploy portlets, and to extend and deploy portal assets.

  You can also develop custom task flows and other custom components for your portal and deploy them to a shared library for use in WebCenter Portal.

- **Round-Trip Development:** Round-trip development refers to features and techniques that allow you to export portal assets from WebCenter Portal and import them back to JDeveloper for maintenance or enhancement. For more information on round-trip development, see "Using Iterative and Round-Trip Development Techniques" in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

- **Enterprise deployment:** When you are ready to deploy your portal to a production environment, you can use the `exportWebCenterPortals` and `importWebCenterPortals` WLST commands. For more information, see Section 40.9.2, "Directly Deploying Portals From Staging to Production" and Section 40.1.2, "Deploying Portal Archives."

  Alternatively, if you want to keep your stage and production environments connected and "in sync " so you can propagate metadata changes from stage to production, use the WLST command `deployWebCenterPortals`. See also, Section 40.1, "Deploying Portals."

- **Browser-based administration tools:** Various browser-based tools enable administrators to configure, manage, and monitor portal deployments:

  - Portal Builder Administration
  - Enterprise Manager Fusion Middleware Control Console
  - WebLogic Server Administration Console

  For more information, see Section 1.13, "Oracle WebCenter Portal Administration Tools". See also, Table 39–3, " WebCenter Portal and WebLogic Server Permission Requirements for Life Cycle Operations".

- **Scripting tools:** Enable system administrators to configure, manage, backup, and restore portal deployments.

  For more information, see Section 1.13.3, "Oracle WebLogic Scripting Tool (WLST)."

39.6 Permissions Required to Perform WebCenter Portal Life Cycle Operations

Table 39–3 describes which WebLogic Server roles and WebCenter Portal permissions are required to perform lifecycle operations.

Table 39–3 WebCenter Portal and WebLogic Server Permission Requirements for Life Cycle Operations

WebCenter Portal Object	Tool	WebLogic Server Role	WebCenter Portal Permission
Portal			
Deploy Portal Directly from WebCenter Portal	WLST	`Monitor` (or higher)	`Portals - Manage All`
Propagate Portal Directly from WebCenter Portal	WLST	`Monitor` (or higher)	`Portals - Manage All`

Table 39–3 (Cont.) WebCenter Portal and WebLogic Server Permission Requirements for Life Cycle

WebCenter Portal Object	Tool	WebLogic Server Role	WebCenter Portal Permission
Import or Export Portal Archive	WLST	`Monitor` (or higher)	`Portals - Manage All`
Import or Export Portal Archive	WebCenter Portal	-	`Portals - Manage All`
Portal Template			
Import or Export Portal Template Archive	WLST	`Monitor` (or higher)	`Portal Templates - Manage All`
Import or Export Portal Template Archive	WebCenter Portal	-	`Portal Templates - Manage All`
Portal Asset			
Import or Export Asset Archive	WLST	`Monitor` (or higher)	Either: ▪ `Create, Edit, Delete Assets` ▪ `Create, Edit, Delete <Portal_Asset_Type>`
Import or Export Asset Archive	WebCenter Portal	-	`Portal - Manage Configuration` And either: ▪ `Create, Edit, Delete Assets` ▪ `Create, Edit, Delete <Portal_Asset_Type>`
Import or Export Asset Archive	JDeveloper	-	-
Portal Extension (ADF shared library)			
Deploy Portal Extension Directly from JDeveloper	JDeveloper	`Monitor` (or higher)	`Portals - Manage All`
Import or Export Portal Extension Archive	JDeveloper	-	-
Portal Connections			
Import or Export Connections	WLST	`Operator` (or higher)	-
Shared Asset			
Import or Export Asset Archive	WLST	`Monitor` (or higher)	`Create, Edit, Delete <Shared_Asset_Type>`
Import or Export Asset Archive	WebCenter Portal	-	`Application - Manage Configuration` `Create, Edit, Delete <Shared_Asset_Type>`
Import or Export Asset Archive	JDeveloper	-	-
WebCenter Portal (all portals)			
Import or Export Archive	WLST	`Monitor` (or higher)	`Application - Manage`
Import or Export Archive	Fusion Middleware Control	`Monitor` (or higher)	`Application - Manage`

39.7 Setting Up a Staging or Production WebCenter Portal Environment for the First Time

If you are setting up stage and production environments for WebCenter Portal for the first time, follow instructions in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*. Once both environments are set up and your stage environment is ready to be migrated to the production server, refer to Chapter 41.5, "Migrating Entire WebCenter Portal to Another Target."

See also Section 39.11, "Managing Security Through the WebCenter Portal Life Cycle" for information on moving security policies and credentials to an environment for the first time.

Note: Cloning an entire *Fusion Middleware* production instance from a stage instance, is sometimes referred to as "test to production". For information on how to clone a WebCenter Portal instance together with other Fusion Middleware components, see "Moving Oracle WebCenter Portal to a New Target Environment," in the *Oracle Fusion Middleware Administrator's Guide*.

39.8 Managing WebCenter Portal Deployment from Your Development Environment

After testing new portal assets and extensions, developers will typically deploy new components to an archive and give them to an administrator for deployment to WebCenter Portal. For example, portal asset archives (`.ear` files), ADF libraries (`.jar` files) and shared libraries (`.war` files).

Developers can deploy portal assets/extensions to WebCenter Portal directly from JDeveloper if given the appropriate permissions to do so. For example, while developers might be allowed to deploy assets/extensions in the test environment but prevented from deploying to stage and production servers. For details, see "Deploying Extensions Directly to the Portal Server" and "Uploading Assets Directly to WebCenter Portal" in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

39.9 Managing Portal Changes in Staging to Production

In practice, the staging environment and the production remain identical until changes are made to the stage environment. You can move approved changes to production in two ways:

- Export updates on stage to a file and then use Portal Builder or WLST commands to import the changes in the file on the production server
- Deploy updates on stage *directly* to production using deploy or propagate WLST commands.

The propagation feature is useful for migrating changes to portal metadata (pages, assets, portlets dropped on to pages, and so on). Propagation does not require the production server to be restarted or incur any downtime because propagation only transfers changes to portal metadata. If you want to use the propagation feature, you must move the first set of changes to production using the `deployWebCenterPortal` command—this command re-creates the entire stage portal on the production instance and creates a matching label on the stage portal and the production portal, for example LABEL_1. Subsequently, when you propagate changes from stage to production the portal's label increments by 1 in both the stage and production environment. You can only propagate portal changes to another portal instance if the labels match.

For more information, see Section 40.9, "Managing Portals in Production."

See also, Appendix E, "Labeling During WebCenter Portal Lifecycle."

39.10 Managing Changes in Production Back Into Staging

You can export individual production portals to an archive and import them back to your staging site, using WLST commands or Portal Builder Administration. For more information, see Section 40.1.2.2, "Deploying Portal Archives to a Different Server."

If you want mirror your entire production system to stage, you have two options; either back up/restore your production WebCenter Portal instance or clone the WebLogic Server that is running your production WebCenter Portal instance. For details, see Chapter 41, "Managing WebCenter Portal Backup, Recovery, and Cloning."

39.11 Managing Security Through the WebCenter Portal Life Cycle

This section discusses techniques for migrating portal security policies and credentials from one WebCenter Portal environment to another.

Security Policy for a Single Portal

Each portal has its own security policy. When you deploy a portal on a WebCenter Portal instance for the first time you must include the portal's security policy. On redeployment, the security policy is optional. For example, if you redeploying a portal from staging to production, often it is important *not* to overwrite policy changes made on the production system. See also, Section 40.1, "Deploying Portals."

Security Policy for an Entire WebCenter Portal Application (all portals, including the Home portal)

When you back up (or export) an entire WebCenter Portal application, security policies for the Home portal and individual portals are included in the archive so you can move/restore the security information on one instance to another. For details, see Section 41.5, "Migrating Entire WebCenter Portal to Another Target."

Back-end Identity Store and Credential Store for WebCenter Portal

When you migrate to another instance, you must migrate the back-end components for security, such as Identity Store, Credential Store, Policy Store. For details, see Section 41.6.7, "Backing Up and Restoring Policy Stores (LDAP and Database)" and Section 41.6.8, "Backing Up and Restoring Credential Stores (LDAP and Database)."

39.12 Managing Backups Through the WebCenter Portal Life Cycle

For more information, see Chapter 41, "Managing WebCenter Portal Backup, Recovery, and Cloning."

40

Deploying Portals, Templates, Assets, and Extensions

WebCenter Portal provides a set of utilities that enable administrators to deploy, back up or move information between WebCenter Portal installations, and stage or production environments.

This chapter includes the following topics:

Permissions: The content of this chapter is intended for system administrators.

For more information on which roles and permissions are required to deploy portals, templates, assets, connections, and extensions, see Section 39.6, "Permissions Required to Perform WebCenter Portal Life Cycle Operations."

See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

40.1 Deploying Portals

This section includes the following topics:

- Section 40.1.3, "Deploying Portal Hierarchies"

40.1.1 About Portal Deployment

When you deploy a portal to another portal server, you make a copy of the source portal on the target server and you can include *all or some* of the source portal's data.

Deploying a new portal on the target server and redeploying an existing portal are exactly the same. When you deploy a portal that already exists on the target, it is simply deleted and recreated as a new portal.

Note: If you want to propagate *metadata changes* on the source portal to the target, rather than redeploying the whole portal, see Section 40.9.3, "Propagating Portal Changes in Staging to Production."

You can deploy online portals to a portal archive or directly deploy portals to another server:

- **Portal archive deployment** - In most situations you will create an archive of the source portal (`.par` file) and then import the archived portal on the target server. You can use Portal Builder administration or WLST commands to export portals to an archive and then import portals from the file.

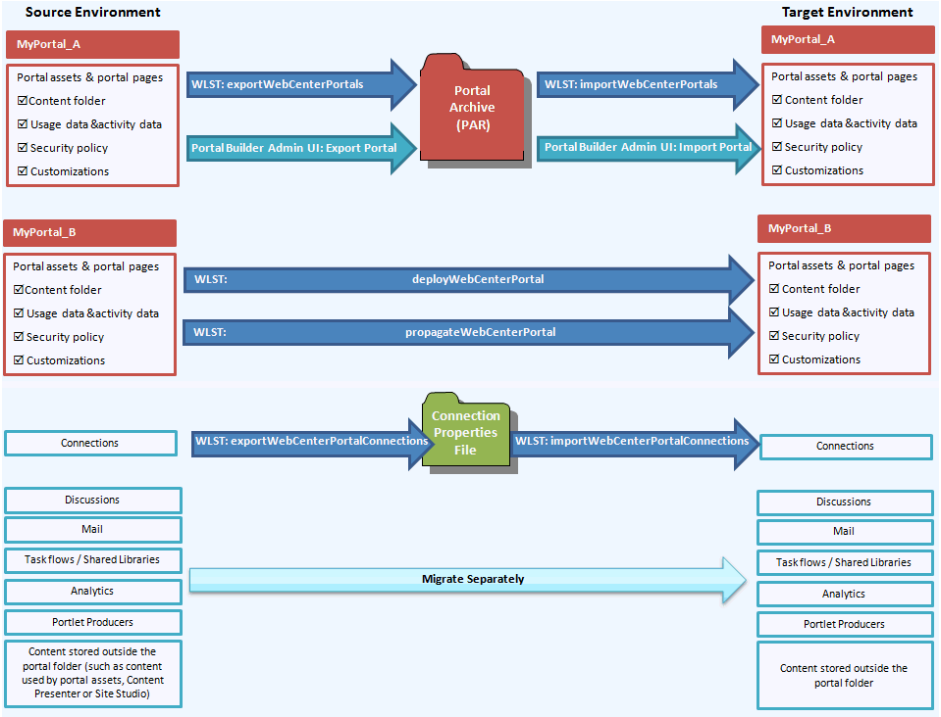
  For details, see Section 40.1.2.2, "Deploying Portal Archives to a Different Server."

- **Direct portal deployment** - If you have a direct connection to your target server, you can use the `deployWebCenterPortal` WLST command to deploy portals to another target server. This method is useful if you want to propagate portal metadata changes.

  For details, see Section 40.9.2, "Directly Deploying Portals From Staging to Production."

Figure 40–1, "Deploying Portals to Another Target Server" illustrates the different ways you can move a portal (and its associated data) to another server.

Figure 40–1 Deploying Portals to Another Target Server



Information Deployed with a Portal

Internally stored data, that is, portal data stored in the WebCenter Portal database, WebCenter Content repository, and MDS can be deployed on the target at the same time as the portal. Depending on your requirements, you can include or exclude this information on deployment by setting options on deployment (or import):

Table 40–1 Portal Deployment Options

Information You Can Deploy with a Portal	WLST Option importWebCenterPortals	Portal Builder Import Option	WLST Option deployWebCenterPortal
Portal pages	Always included	Always included	Always included
Assets ▪ Page templates ▪ Navigations ▪ Resource catalogs ▪ Skins ▪ Page styles ▪ Content presenter display templates ▪ Task flow styles ▪ Task flows ▪ Data control ▪ Pagelets	Always included	Always included	Always included
Portal content folder	importPortalContent	Include Portal Folder on Content Server	deployPortalContent

Table 40–1 (Cont.) Portal Deployment Options

Information You Can Deploy with a Portal	WLST Option importWebCenterPortals	Portal Builder Import Option	WLST Option deployWebCenterPortal
Portal activity/usage data: ■ activity streams ■ calendar events ■ feedback ■ lists ■ links ■ message boards ■ people connections ■ polls ■ profiles ■ surveys	importData	Include Services Data	deployData
Portal security data: (Optional if portal exists on the target) ■ portal roles and permissions ■ member details and their role assignments	importSecurity	Include Security	deploySecurity
Portal customizations: ■ portal-level customizations (system pages, assets, task flows, and so on) ■ user-level customizations (sometimes referred to as user personalizations)	importCustomizations	Include Customizations	deployCustomizations

Information Not Migrated During Portal Deployment

Some portal information is stored externally and cannot be deployed at the same time as the portal, for example:

- content used by portal assets, Content Presenter or Site Studio
- portal discussions
- portal mail
- portal analytics
- custom task flows / shared libraries
- portal connections *

Other information not migrated includes:

- shared assets
- Home portal

Note: * Connections are exported and imported separately. For more information, see Section 40.1.2.1.3, "Understanding Connection Property Files."

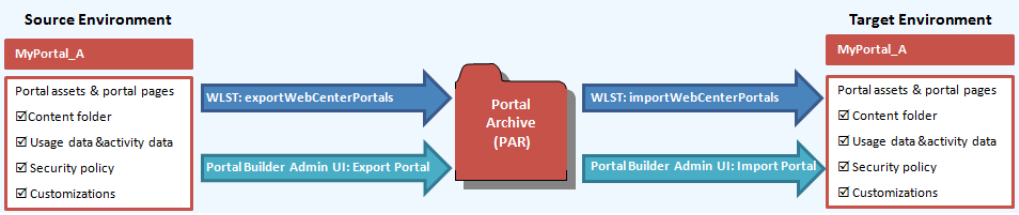
If your source and target WebCenter Portal installations are connected to different external servers and information associated with the source portal is required on the target, the external portal data must be moved separately. For details, see Section 40.8.1, "Migrating Back-end Components for Individual Portals."

In some situations the source and target both use the same external server, for example, a portlet producer server or Oracle Internet Directory server might be shared across both environments.

40.1.2 Deploying Portal Archives

Administrators can use Portal Builder or WLST commands to deploy portal archives (`.par` files) to any WebCenter Portal (11.1.1.8.x) installation. The target WebCenter Portal must be up and running when you deploy (or import) one or more portals from a file.

Figure 40–2 Deploying Portal Archives



This section includes the following topics:

- Section 40.1.2.1, "Understanding Portal Archives"
- Section 40.1.2.2, "Deploying Portal Archives to a Different Server"
- Section 40.1.2.3, "Creating Portal Archives"
- Section 40.1.2.4, "Importing One or More Portals from an Archive"
- Section 40.1.2.5, "Viewing and Extracting Portal Archives"

> **Note:** When you deploy a portal to another server from an archive you cannot use portal propagation to make incremental updates to the portal later on. The portal propagation feature is only possible when used in conjunction with direct portal deployment. See Section 40.9, "Managing Portals in Production."

40.1.2.1 Understanding Portal Archives

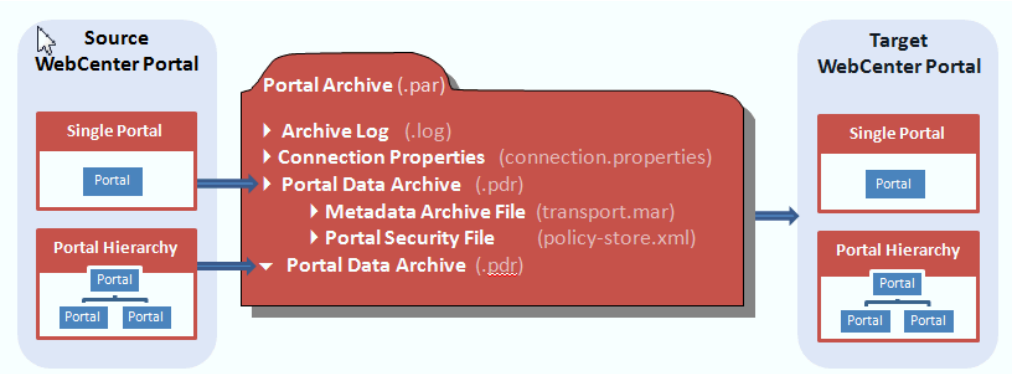
You can use Portal Builder or the `exportWebCenterPortals` WLST command to create a portal archive (`.par` file) for a single portal, a complete portal hierarchy, or you can archive multiple portals and portal hierarchies in the same `.par` file.

Portal archives can contain:

- One or more portal data archive files (`.pdr`)
- An export log file (`.log`)
- A WebCenter Portal connection properties file (`connection.properties`)

Note: You can extract any portal archive (`.par` file) using the `listWebCenterPortalArchive` WLST command.

Figure 40–3 Portal Archive Deployment



This section includes the following:

- Understanding Portal Data Files (PDRs)
- Understanding Export Log Files
- Understanding Connection Property Files

40.1.2.1.1 Understanding Portal Data Files (PDRs)

Portal archives include a portal data file (PDR) for each portal or portal hierarchy that you add to the archive, as illustrated in Figure 40–3, "Portal Archive Deployment".

The portal data file contains two files:

- `transport.mar` - A metadata archive that captures metadata, data, and content for a single portal or portal hierarchy:
  - **Metadata**: Metadata stored in MDS for portal pages, assets, global customizations, user customizations, portlets, and so on.
  - **Data**: Data stored in WebCenter Portal database tables for portal activity, portal events, and so on.
  - **Content**: Content in the folder specifically created for any portal that offers document services.

    The folder is archived to a `.zip` file located at: `oracle\webcenter\lifecycle\importexport\data\oracle-webcenter-doclib\docsexport.zip`

    The archive **does not** include any web content that is referenced by the portal or content that is stored at any other location. Only the folder assigned to the portal on the back-end WebCenter Content Server is included with the portal.
- `policy-store.xml` - Security policy for the portal or portal hierarchy.

Table 40–2, " Information Exported to Portal Archives (PDR Files)" describes the information included in `.pdr` file in more detail, shows you how to where information is located within the file, and highlights which information is optional on import.

Table 40–2 Information Exported to Portal Archives (PDR Files)

Information Exported to PDR Files	Exported	Imported	Location in PDR
	Always / Optional / Never		
Portal pages	Always	Always	`\transport.mar\oracle\webcenter\page\`
Portlets	Always	Always	`\oracle\adf\portlet`
Portal Assets ■ Page templates ■ Navigations ■ Resource catalogs ■ Skins ■ Page styles ■ Content presenter display templates ■ Task flow styles ■ Task flows ■ Data controls ■ Pagelets	Always	Always	`\transport.mar\oracle\webcenter\siteresources\`
Portal content folder	Optional	Optional	`\transport.mar\oracle\webcenter\lifecycle\importexport\data\oracle-webcenter-doclib\docsexport.zip`
Portal activity/usage data: ■ activity streams ■ calendar events ■ feedback ■ lists ■ links ■ message boards ■ people connections ■ profiles ■ polls ■ surveys ■ tags	Always	Optional	`\transport.mar\oracle\webcenter\lifecycle\importexport\data`
Portal security data: ■ portal roles and permissions ■ member details and their role assignments	Always	Optional (If portal exists on the target)	`policy-store.xml`
Portal customizations: ■ portal-level customizations (system pages, assets, task flows, and so on) ■ user-level customizations (sometimes referred to as user personalizations)	Always	Optional	`\transport.mar\oracle\webcenter\`

Table 40–2 (Cont.) Information Exported to Portal Archives (PDR Files)

Information Exported to PDR Files	Exported	Imported	Location in PDR
	Always / Optional / Never		
External content referenced by the portal through: ■ portal pages ■ portal assets ■ Content Presenter display templates ■ Site Studio ■ ...and so on	Never*	-	-
Data stored on external servers: ■ discussions ■ mail ■ announcements ■ analytics ■ custom task flows and shared libraries	Never*	-	-
Shared assets	Never*	-	-
Home portal	Never*	-	-

***Information Not Included in PDR Files**

Portal data archives do not include shared assets or any information relating to the Home portal.

Portal data archives do not include content that is stored outside the portal's own content folder, such as content used by portal assets, portal pages, Content Presenter or Site Studio. Similarly, the archive does not contain other externally stored data such as discussions, announcements, and mail. To learn how to move this data, see Section 40.8.1, "Migrating Back-end Components for Individual Portals."

40.1.2.1.2 Understanding Export Log Files

When you export one or more portals, a log file (`.log` file) is provided inside the portal archive (`.par` file). The export log lists all the portals, MDS metadata files, and data (names of database tables that contain portal data) that are included in the archive.

Export logs are very useful for reference purposes and if you want to compare portal archives.

Example 40–1, "Portal Archive Export Log" shows the general format of an export log.

Example 40–1 Portal Archive Export Log

```
Exporting portal: MySalesPortal

Processing subportals : MySales, TeamSales

MDS Documents included for portal: MySalesPortal, MySales, TeamSales
/oracle/webcenter/lifecycle/importexport/data.xml
/oracle/webcenter/siteresources/...
```

```
/oracle/webcenter/page/...
/pageDefs/oracle/webcenter/page/...
/oracle/webcenter/siteresources/...
/oracle/webcenter/space/metadata/spaces/...
/oracle/webcenter/translations/...
/oracle/webcenter/space/metadata/spaces/...
/oracle/webcenter/security/...
/oracle/webcenter/framework/scope/...

Exported database data for portal: MySalesPortal, MySales, TeamSales  at
2012-09-26-05:00:59

Database data included for portal: MySalesPortal, MySales, TeamSales
  WC_LIST_ROW$
  WC_RELATIONSHIP_RESOURCE
  WC_RELATIONSHIP_RESOURCE_ATT
  WC_RELATIONSHIP_INFO
  WC_RELATIONSHIP
  WC_RELATIONSHIP_INFO_ATT
  WC_PPL_COMMON_SETTING
  WC_SURVEY
  WC_SURVEY_PROPERTY
  WC_SURVEY_QUESTION
  WC_SURVEY_QUESTION_PROPERTY
  WC_CAL_CATEGORY
  WC_CAL_EVENT
  WC_CAL_ATTENDEE
  WC_AS_ACTIVITY_ELEMENT
  WC_AS_CUSTATTR
  WC_AS_OBJECT_DETAIL
  WC_AS_OBJECT
  WC_AS_ACTOR_DETAIL
  WC_AS_ACTOR
  WC_COMMENT
  WC_LIKE

Archive created for portal: MySalesPortal, MySales, TeamSales.

Portal data archive created:
/tmp/expportal7553974727669501601/MySalesPortal-2012-09-26-05:00:59.pdr
```

40.1.2.1.3 Understanding Connection Property Files

When you export portals to an archive using the WLST command `exportWebCenterPortals`, connection information used in the source portal environment to connect to portlet producers, web services, external servers, and applications is saved to a file named `connection.properties`. The `connection.properties` file is included inside the portal archive and a copy is also placed in same directory as the portal archive.

Connection information is included with the portal archive for backup purposes only, so you can easily see the connection configuration at the time the portal was backed up and if necessary, compare it with connection configuration at another point in time or perhaps on a restored WebCenter Portal instance.

If you plan to import, deploy, propagate, or restore a portal on a WebCenter Portal target where all or some connections do not exist, Oracle recommends that you use the WLST command `exportWebCenterPortalConnections` to generate the `connection.properties` file from the source environment, and then use the WLST command `importWebCenterPortalConnections` to import missing connections

configured in that file on the target environment. For detailed steps, see Section 40.6.2, "Importing New WebCenter Portal Connections from a File."

Notes:

- A `connection.properties` file is generated when you export a portal using the WLST command `exportWebCenterPortals` but this file is not included when you export portals from Portal Builder Administration.
- A `connection.properties` file is also generated when you run the WLST command `exportWebCenterPortalConnections`. For details, see, Section 40.6.1, "Exporting WebCenter Portal Connections Details to a File."
- All connections configured in the source WebCenter Portal environment are exported to `connection.properties`. The connection information in this file is not specific to the portals in the archive.
- Only new connections are imported on the target. Connections that exist on the target are ignored.

Modifying Connection Details

If some connection information, such as server names, ports, and so on, varies between the source and target environments, you can isolate and modify connection details in the file before importing, deploying, propagating, or restoring the portal.

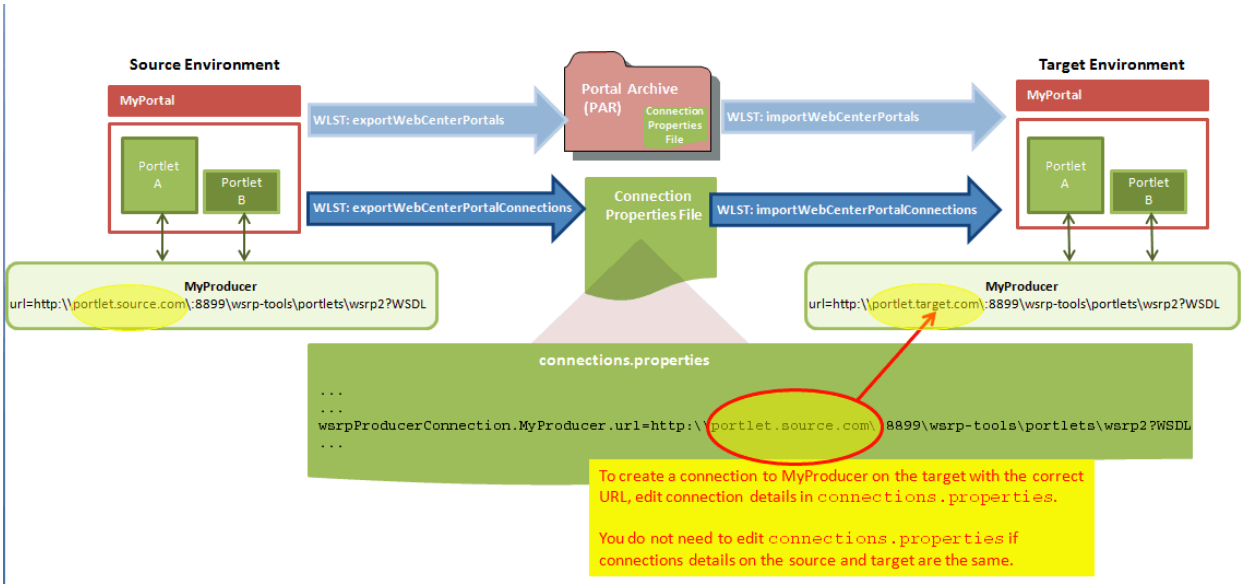
Table 40–3 shows examples where a different URL parameter is required on the target because the source and target do not use the same host.

Table 40–3 Example: Connection URLs Different in Source and Target Environments

Connection Type	Source Connection: URL parameter	Target Connection: URL parameter
WSRP Portlet Producer	http://**mysource.com**:8899/MyWSRPPortletProducer/portlets/wsrp2?WSDL	http://**mytarget.com**:8899/MyWSRPPortletProducer/portlets/wsrp2?WSDL
PDK-Java Producer	http://**source.host.com**:7778/myJPDKPortletProducer/providers	http://**target.host.com**:7778/myJPDKPortletProducer/providers
Web Service	http://**source.example.com**/getEmployee?empId=20+deptId=10	http://**target.example.com**/getEmployee?empId=20+deptId=10

Figure 40–4 illustrates how you can edit connection details in `connection.properties` when source and target parameters vary *before* new connections are created on the target.

Figure 40–4 Using connection.properties to Create Connections on the Target



Connection Types and Connection Properties

Table 40–4 lists all the connections captured in the `connection.properties` file together with the properties that are exported for the various connection types. The table also shows which properties you can edit before deployment, and which properties you must set on the target.

Note:

- For detailed information about individual connection properties, including which ones are mandatory or optional for a particular connection type, refer to the chapter for that connection type. For a list of chapters, see Part V, "Managing Tools, Portlet Producers, and External Applications".
- Oracle strongly recommends that you only edit properties in `connection.properties` that are marked **Edit on Deployment?**=**Yes** in Table 40–4. If for some reason you want to edit any of the properties marked **Edit on Deployment?**=**No**, you may do so after migrating the connection on the target using either Fusion Middleware Control or WLST commands.

Table 40–4 Connection Properties Exported to connection.properties

Connection Type	Properties Exported	Edit on Deployment?	Notes and Post Deployment Configuration Requirements

WSRP portlet producer	`url`	Yes	Security configuration post deployment:
			`registrationProperties`
			`keyStorePath`
			`keyStorePswd`
			`sigKeyAlias`
			`sigKeyPswd`
			`encKeyAlias`
			`encKeyPswd`
			`enforcePolicyURI`
			[1]

	`proxyHost`	Yes	
	`proxyPort`	Yes	
	`timeout`	No	
	`externalApp`	No	
	`tokenType`	No	
	`defaultUser`	No	
	`issuerName`	No	
	`recipientAlias`	No	

PDK-Java producer	`url`	Yes	Security configuration post deployment:
			`mapUser`
			`useProxy`

	`proxyHost`	Yes	
	`proxyPort`	Yes	
	`subscriberId`	No	
	`serviceId`	No	
	`sharedKey`	No	
	`timeout`	No	
	`establishSession`	No	
	`externalApp`	No	
Web service connection	`url`	Yes	Web service connections are used by data controls [1]
	`proxyHost`	Yes	
	`proxyPort`	Yes	
	`mtom`	No	
	`addressing`	No	
	`wsrm`	No	
	`security`	No	

URL connections - HTTP URL	`url`	Yes	Security configuration post deployment:
			`password`
			`user`
			`attributes`

	`authenticationType`	No	
	`connectionClassName`	No	
	`realm`	No	
URL connections - File URL	`url`	Yes	
Pagelet producers	`url`	Yes	
Analytics collector	`collectorPort`	Yes	`host`: represents `clusterName` when `isUnicast` is set to 0 and `collectorHost` when `isUnicast` is set to 1
	`host`	No	
	`isEnabled`	No	
	`timeout`	No	
	`isUnicast`	No	
	`defaultConnection`	No	

Table 40–4 (Cont.) Connection Properties Exported to connection.properties

Connection Type	Properties Exported	Edit on Deployment?	Notes and Post Deployment Configuration Requirements
BPEL server	url	Yes	
	policy	No	
	recipientKeyAlias	No	
	linkURL	No	
Discussions server	url	Yes	
	adminUser	Yes	
	application.root.category.id	Yes	
	recipientKeyAlias	No	
	policyURIForAuthAccess	No	
	policyURIForPublicAccess	No	
	timeout	No	
	defaultConnection	No	
External applications	url	Yes	If public or shared credentials are configured on the source, they are not exported for security reasons. You must configure these credentials on the target post deployment, if required.
	authMethod	No	
	userFieldName	No	
	pwdFieldName	No	
	displayName	No	
	publicCredentialEnabled	No	
	sharedCredentialEnabled	No	
	AdditionalFields	No	
Presence server - Microsoft Live Communications Server	url	Yes	
	poolName	Yes	
	adapter	No	
	timeout	No	
	appId	No	
	AdditionalProperty	No	
	defaultConnection	No	
Presence server - Microsoft Office Communications Server 2007 / Microsoft Lync 2010	url	Yes	
	poolName	Yes	
	userDomain	Yes	
	adapter	No	
	timeout	No	
	appId	No	
	AdditionalProperty	No	
	defaultConnection	No	

Table 40–4 (Cont.) Connection Properties Exported to connection.properties

Connection Type	Properties Exported	Edit on Deployment?	Notes and Post Deployment Configuration Requirements
Mail server	imapHost	Yes	LDAP configuration post deployment: LdapDomain LdapDefaultUser LdapHost LdapBaseDn LdapAdminUsername LdapPort LdapSecured
	smtpHost	Yes	
	imapPort	Yes	
	smtpPort	Yes	
	smtpSecured	Yes	
	imapSecured	Yes	
	appId	No	
	timeOut	No	
	AdditionalProperties	No	
	defaultConnection	No	
Personal events server	webServiceUrl	Yes	
	adapterName	No	
	appId	No	
	defaultConnection	No	
Oracle SES	url	Yes	Users will be prompted for appPassword if promptForPassword is set to 1
	appUser	No	
	defaultConnection	No	
WebCenter Content Server (socket)	serverHost	Yes	Security configuration post deployment: adminUsername adminPassword keystorePassword privateKeyPassword
	serverPort	Yes	
	extAppId	No	
	timeout	No	
	socketType	No	
	webContextRoot	No	
	cacheInvalidationInterval	No	
	binaryCacheMaxEntrySize	No	
	defaultConnection	No	
WebCenter Content Server (socketssl)	serverHost	Yes	Security configuration post deployment: adminUsername adminPassword keystorePassword privateKeyPassword
	serverPort	Yes	
	extAppId	No	
	timeout	No	
	socketType	No	
	webContextRoot	No	
	cacheInvalidationInterval	No	
	binaryCacheMaxEntrySize	No	
	defaultConnection	No	
	keystoreLocation	No	
	privateKeyAlias	No	

Table 40–4 (Cont.) Connection Properties Exported to connection.properties

Connection Type	Properties Exported	Edit on Deployment?	Notes and Post Deployment Configuration Requirements
WebCenter Content Server (jaxws)	url	Yes	Security configuration post deployment: adminUsername adminPassword keystorePassword privateKeyPassword
	extAppId	No	
	timeout	No	
	socketType	No	
	webContextRoot	No	
	cacheInvalidationInterval	No	
	binaryCacheMaxEntrySize	No	
	defaultConnection	No	
	clientSecurityPolicy	No	
WebCenter Content Server (web)	url	Yes	Security configuration post deployment: adminUsername adminPassword keystorePassword privateKeyPassword
	extAppId	No	
	timeout	No	
	socketType	No	
	webContextRoot	No	
	cacheInvalidationInterval	No	
	binaryCacheMaxEntrySize	No	
	defaultConnection	No	
Oracle Portal	dataSource	Yes	
	extAppId	No	
	timeout	No	
File System	path	Yes	
Worklist connection	BPELConnection	No	

[1] **Security related configuration:** Only policy information is included with the connection. The *Override* set for the security policy is not included so you must configure these parameters post deployment.

To find out how to deploy connection information in to another server, see Section 40.6, "Moving Connections Details from Staging to Production."

40.1.2.2 Deploying Portal Archives to a Different Server

First you deploy (or export) the portal to an archive (`.par` file) and then you import the archive on the target server. See also, Section 40.1.2.1, "Understanding Portal Archives."

> **Note:** When you deploy a portal to another server from an archive you cannot use portal propagation to make incremental updates to the portal later on. The portal propagation feature is only possible when used in conjunction with direct portal deployment. See Section 40.9, "Managing Portals in Production."

To export and then import a portal archive:

1. Complete the portal archive prerequisites described in Section 40.1.2.3.1, "Portal Archiving Prerequisites."
2. Export the source portal:

- To use WLST, see Section 40.1.2.3.3, "Exporting Online Portals to an Archive Using WLST."
- To use Portal Builder, see Section 40.1.2.3.2, "Exporting Online Portals to an Archive Using Portal Builder Administration."

3. (Optional) Migrate externally stored data and content to the target:

   For details, see Section 40.8.1, "Migrating Back-end Components for Individual Portals."

4. Import the portal on the target:
   - To use WLST, see Section 40.1.2.4.3, "Importing a Portal from an Archive Using WLST."
   - To use Portal Builder, see Section 40.1.2.4.2, "Importing a Portal from an Archive Using Portal Builder Administration."

40.1.2.3 Creating Portal Archives

Administrators can generate an archive (`.par` file) for any portal or portal hierarchy that is running on WebCenter Portal. You can use Portal Builder or the WLST command `exportWebCenterPortals` to create the archive and optionally include the portal's content folder.

To find out how to create portal archives, see:

- Section 40.1.2.3.1, "Portal Archiving Prerequisites"
- Section 40.1.2.3.2, "Exporting Online Portals to an Archive Using Portal Builder Administration"
- Section 40.1.2.3.3, "Exporting Online Portals to an Archive Using WLST"

40.1.2.3.1 Portal Archiving Prerequisites

Before exporting a portal to an archive (`.par` file), verify the following:

- **Web service data controls -** If any of the portals you want to export contain web service data controls, all the associated web services must be up and accessible for the export to succeed.
- **Portlet producers -** If any of the portals you want to export contain portlets, all associated portlet producers must be up and accessible for all portlet metadata to be included in the archive.
- **Content outside portal folder** - Content stored outside the portal folder (such as files, images and icons) that is used by portal assets, portal pages, Content Presenter, and Site Studio are not automatically included in the archive. You must copy all dependent files to appropriate locations on the target content server.

Note: If you are managing legacy portals with assets that store artifacts in MDS, Oracle recommends that you relocate all dependent artifacts from MDS to your content server. If you choose not to move artifacts stored in MDS, you can use MDS WLST commands `exportMetadata`/`importMetadata` to move the MDS content to another target. For example:

```
exportMetadata(application='webcenter', server='WC_Spaces',
 toLocation='/tmp/content',
 docs='/oracle/webcenter/siteresources/scopedMD/shared/**')

importMetadata(application='webcenter', server='WC_Spaces',
 fromLocation='/tmp/content',
 docs='/oracle/webcenter/siteresources/scopedMD/shared/**')
```

40.1.2.3.2 Exporting Online Portals to an Archive Using Portal Builder Administration

WebCenter Portal administrators can export portals to an archive using Portal Builder administration as described here. You can save the portal archive to your local file system or to a remote server file system.

Note: You can export portal templates too, but this is a separate process. You cannot export portals and portal templates into a single archive. For details, see Section 40.2.1.1, "Exporting Portal Templates to an Archive Using WebCenter Portal."

To export one or more portals through Portal Builder administration:

1. Click the **Administration** link.
2. Click the **Portals** tab to display the **Portals** page.

   You can also enter the following URL in your browser to navigate directly to the **Portals** management page:

   `http://`*`host`*`:`*`port`*`/webcenter/portal/builder/portals`

   See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*

3. Select the portal required by highlighting the row in the table.

   Ctrl-click rows to select more than one.

   Tip: to prevent data conflict during the export process, Oracle recommends that all the portals you select (including subportals) are *offline* during the export process, even if only temporarily. For details, see Section 46.11, "Taking Any Portal Offline."

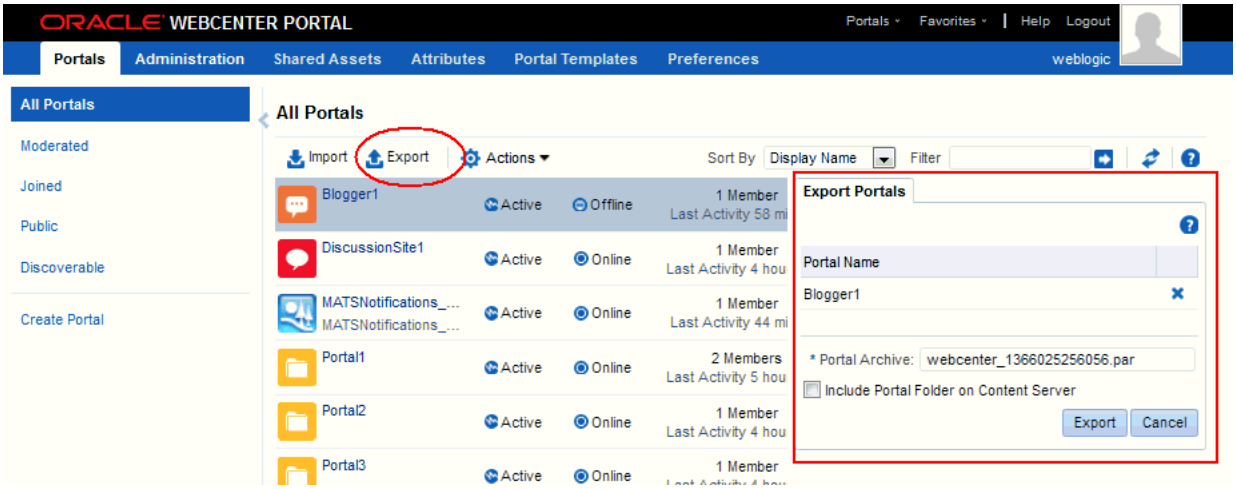
   Members with the `Portals-Manage All` permission can still access a portal when it is offline, so ask them not make changes while you do the export.

4. Click **Export** in the toolbar.

The Export Portals pane opens (Figure 40–5). All the portals that you select are listed. If you chose a portal hierarchy, you must export the entire hierarchy; you cannot selectively remove child portals.

If you want to exclude a portal, click the **Delete** icon next to the portal's name (a blue cross in Figure 40–5).

Figure 40–5 Exporting Portals



5. Enter a name for the **Portal Archive** with the file extension `.par` or accept the default name.

   The default filename for the portal archive includes a random number to ensure uniqueness: `webcenter_`*`random_number`*`.par`

6. Select **Include Portal Folder on Content Server** to export each portal's content folder.

   A folder is automatically created in WebCenter Portal's content repository for portals that use document services to create, manage, and store portal documents (files, folders, wikis, blogs). Only content that is stored in this folder can be exported with the portal. The export does not, for example, include web content/pages displayed through Content Presenter since this information is not stored in the portal's content folder. See also, Section 40.1.2.1.1, "Understanding Portal Data Files (PDRs)."

Notes:

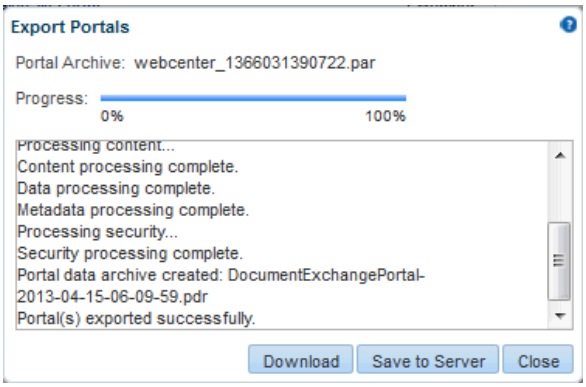
- Including content folders increases the size of the portal archive. If you are exporting a large number of portals or large content folders, take care that your archive does not exceed the maximum upload size for files (2 GB by default). If necessary, you can increase this setting, as described in Section 9.12, "Changing the Maximum File Upload Size."
- If you are managing legacy portals with assets that store artifacts in MDS, Oracle recommends that you relocate all dependent artifacts from MDS to your content server. If you choose not to move artifacts stored in MDS and do not include MDS content within the asset archive, you can use MDS WLST commands `exportMetadata`/`importMetadata` to move the MDS content another time. For example:

```
exportMetadata(application='webcenter', server='WC_Spaces',
 toLocation='/tmp/content',
 docs='/oracle/webcenter/siteresources/scopedMD/shared/**')

importMetadata(application='webcenter', server='WC_Spaces',
 fromLocation='/tmp/content',
 docs='/oracle/webcenter/siteresources/scopedMD/shared/**')
```

7. Click **Export**.
8. Progress information is displayed during the export process (Figure 40–6).

   When the export process is complete, specify a location for the export archive (`.par` file).

Figure 40–6 Portal Export In Progress



Select one of:

- **Download** - Saves the export `.par` file to your local file system.

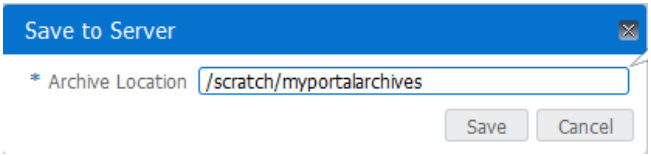
  Your browser downloads and saves the archive locally. The actual download location depends on your browser settings.

  Some browsers have settings that restrict the size of downloads. If your export archive is large and does not download, check your browser settings.

- **Save to Server** - Saves the export `.par` file to a server location.

When the Save to Server dialog displays (Figure 40–7), enter a suitable path in **Archive Location**, for example, `/tmp`, and then click **Save**. Ensure that the server directory you specify exists and you have `write` permissions.

Figure 40–7 Saving Export Archives to a Server Location



9. Click **Close** to dismiss the Export Portals pane.

The export archive (`.par` file) is saved to the specified location.

40.1.2.3.3 Exporting Online Portals to an Archive Using WLST

Use the WLST command `exportWebCenterPortals` to export one or more portals to a portal archive (`.par` file). When you create a portal archive using WLST you can choose whether or not to include the portal's content folder and connection information in the archive:

```
exportWebCenterPortals(appName, fileName, [names, offlineDuringExport,
 exportPortalContent, exportConnections, server, applicationVersion])
```

The options that you set depends on your specific archive requirements. For command syntax, see the "exportWebCenterPortals" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Here are a few examples:

Example 1 - Exporting two portals

This example exports two portals named `Sales` and `Finance`, plus all content, data, security, customizations, and connection information:

```
exportWebCenterPortals(appName='webcenter', fileName='MyPortalExport.par',
 names='Sales,Finance', exportPortalContent=1, exportConnections=1)
```

Example 2 - Exporting a single, offline portal without its content folder or connection details

This example takes `MySales` offline and exports the portal to `MyPortalExport.par`:

```
exportWebCenterPortals(appName='webcenter', fileName='MyPortalExport.par',
 names='Sales', offlineDuringExport=1)
```

40.1.2.4 Importing One or More Portals from an Archive

Administrators can deploy archived portals (`.par` files) to any WebCenter Portal Server. You can use the WLST command `importWebCenterPortals` to import portal archives or you can use Portal Builder administration.

On import, *all* portals included in the archive are created or re-created on the target server. Existing portals are deleted then replaced, and new portals are created. If you intend to import portals with names identical to those available on the target server, ensure that those portals are *offline* in the target application as it is not possible to

overwrite portals that are online. For details, see Section 46.11, "Taking Any Portal Offline."

> **Note:** When importing portals using WLST, you can set the option `forceOffline=1` to automatically take any online portals offline. Any portals taken offline in this way, remain offline at the end of the import process.

Portals are locked during an import operation to prevent simultaneous imports/exports of the same portal. If someone else is importing a particular portal, all subsequent attempts to import (or export) the same portal are blocked.

After importing one or more portals, consider initiating an Oracle Secure Enterprise Search crawl to index the newly imported data.

Portal Security (Optional on Import)

All portals need a security policy to work properly so you must include the portal's security policy when you import a brand new portal for the first time. Existing portals have a security policy in place so, in this case, it's up to you whether to overwrite the security information on import or maintain the existing security policy.

Take particular care when importing security on a production instance as you will overwrite the existing security model.

If you choose to import security for an existing portal, the security policy updates do not apply immediately. Any user that is logged in to WebCenter Portal must log out and log back in to adopt the latest security policies for the portal.

Take care when importing and overwriting security on a production instance.

> **Note:** The users in both the source and target environments must be identical. If a shared identity store is not used, your system administrator must migrate users to the target. Refer to Section 41.6.1.2, "Back Up (Export) WebCenter Portal Schema Data" and Section 41.6.1.3, "Restore (Import) WebCenter Portal Data."

Portal Archive Data (Optional on Import)

If data migration is important, you can elect to import data associated with the source portal such as activity streams, events, feedback, lists, links, message boards, people connections, profiles, polls, and surveys.

In some cases this data is not required on the target, for example, when importing portals between a stage and production environments.

Portal Archive Content (Optional on Import)

Portal archives sometimes contain the portal's content folder. If included, you can choose whether or not to import this information too. On import, the content folder in the archive overwrites the folder on the target (if one exists).

> **Note:** Portal archives do not include web content/pages displayed through Content Presenter since this information is not stored in the portal's content folder.

External Portal Data (Import Separately)

Externally stored data, such as discussions can be migrated for individual portals but this is a separate process. See Section 40.8.1, "Migrating Back-end Components for Individual Portals."

To find out how to import portal archives, see:

- Section 40.1.2.4.1, "Portal Import Prerequisites"
- Section 40.1.2.4.2, "Importing a Portal from an Archive Using Portal Builder Administration"
- Section 40.1.2.4.3, "Importing a Portal from an Archive Using WLST"

40.1.2.4.1 Portal Import Prerequisites

Before importing a portal archive (`.par` file), verify the following:

- **Shared identity store** - Verify that the users in the source and target environments are the same.
- **Portals exist on the target** - Check whether any portals in the archive exist on the target. If required, take existing portals offline during the import process.
- **Web service data controls -** If any of the portals you want to import contain web service data controls, all the associated web services must be up and accessible for the import to succeed.
- **Portlet producers** - Any portlet producers used by the portal must be up and running when you import the portal.
- **Shared assets** - If any of the portals reference shared assets, ensure that you move the required shared assets to the target before (or after) the portal is imported to ensure the portal works properly.
- **Connections to external servers, applications, web services, and portlet producers -** Portals that rely on certain external connections to be configured will not work if a similar connection does not exist in the target. Before importing the portal, ensure that all the required connections exist on the target. If you create or reconfigure connections on the target you may need to restart the target managed server. For details, see Section 40.6, "Moving Connections Details from Staging to Production."
- **Archive version -** Verify that the portal archive or portal template archive that you want to import was exported from WebCenter Portal 11.1.1.8.0 or later. Archives from these releases have the `.par` file extension.

  While you can import archives from earlier versions (with the `.ear` file extension), any portal or portal template that you import this way will not fully function on an upgraded WebCenter Portal 11.1.1.8.x. instance. Oracle recommends that you upgrade your source environment to 11.1.1.8.x and then create export archives (`.par` files). For details, see the "Patching Oracle WebCenter Portal" chapter in *Oracle Fusion Middleware Patching Guide*.

40.1.2.4.2 Importing a Portal from an Archive Using Portal Builder Administration

WebCenter Portal administrators can import one more portals and portal hierarchies from a portal archive through Portal Builder administration.

To import one or more portals from a `.par` file:

1. Click the **Administration** link.

2. Click the **Portals** tab to display the **Portals** page.

   You can also enter the following URL in your browser to navigate directly to the **Portals** management page:

   `http://host:port/webcenter/portal/builder/portals`

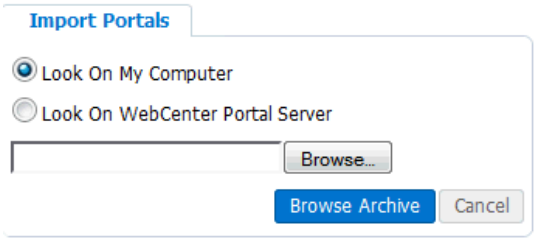
   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*

3. Click **Import** in the toolbar.

   > **Tip:** To import one or more portals *as a subportal* of an existing portal, select the target portal *before* clicking the **Import** button.

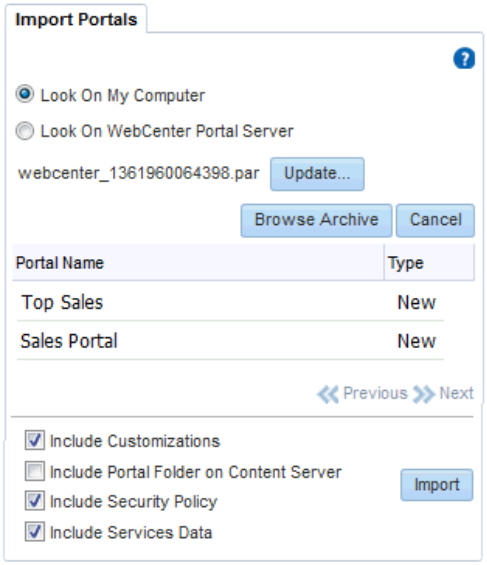
   The Import Portals dialog opens (Figure 40–8).

Figure 40–8 Importing Portals



4. Specify the location of your portal archive (`.par` file). Select one of:
   - **Look on My Computer** - Enter the location in the text box. Alternatively, click **Browse** to locate the directory on your local file system where the `.par` file is stored.
   - **Look on WebCenter Portal Server** - Enter the path on the server where WebCenter Portal is deployed, including the archive filename, in the text box. For example, `/tmp/MyPortalExport.par`. You can specify any shared location accessible from WebCenter Portal.
5. Click **Browse Archive** to review the content available for import (Figure 40–9).

Figure 40–9 Importing Portals



The names of all the portals in the specified archive display in the table. The **Type** column indicates when there is a conflict between the portals in the archive and those which exist on the target:

- **New** - A portal with this name does not exist on the target. On import, a new portal is created.
- **Replace** - A portal with this name and the same GUID exists on the target. The existing portal is deleted on import and replaced with the version in the portal archive.
- **Conflict** - A portal with this name exists on the target but the portal on the target has a different GUID to the portal you are trying to import. Or similarly, this portal has the same GUID as one of the portals in the target but the portal names do not match.

  If the import process detects a conflict between the portals you are trying to import and those which exist on the target, you must resolve the issue. For example, if the conflict is due to matching names but different GUIDs you could either change the name of the source portal and create a new export archive, or rename the conflicting portal in the target application and import the same archive.

6. Set import options as required. For details, see Table 40–5:

Table 40–5 Portal Import Options

Field	Description
Include Customizations	Select to import portal customizations. For information about which customizations are optional on import, refer to Table 40–1, " Portal Deployment Options" and Table 40–2, " Information Exported to Portal Archives (PDR Files)". Import does not completely overwrite existing portal customizations on the target (if any). Only customizations included in the portal archive are overwritten on import leaving any other customizations on the target unchanged. If you deselect this option: ▪ New portals are imported without customizations, for example, default task flows are imported without any customizations, default portal settings are used, and all user customizations are excluded. ▪ If you are importing a portal that already exists on the target, existing customizations on the target are preserved. Note: Portlet and page customizations are always imported.
Include Portal Folder on Content Server	(Only displays if the archive specified includes a content folder for one or more portal.) Select to import all content folders included in the archive. Folders that exist on the target are overwritten on import. Deselect this option to exclude portal content folders (if any). This option is useful when migrating between stage and production environments where test content is no longer required. **Note:** Portal archives that contain large content folders may exceed the maximum upload size for files (2 GB by default). Oracle recommends that you use the `importWebCenterPortals` WLST command to import any portal archive that exceeds the current upload size. See Section 40.1.2.4.3, "Importing a Portal from an Archive Using WLST." If necessary, you can increase the upload setting, see Section 9.12, "Changing the Maximum File Upload Size."
Include Security Policy	Select to import security information with the. portal. When selected, the following security related information is imported: ▪ Portal roles (and permissions assigned to each role). ▪ Portal members (and member role assignments). Deselect this option if you do not want to import portal security information. This option is useful when importing portals between a stage and production environments, where: ▪ Members used during testing are not required in the production environment. ▪ The portal exists on the production instance and you do not want to overwrite the security information. **Note:** When importing a brand new portal, always select (check) this option as you cannot import a new portal without a security policy.

Table 40–5 (Cont.) Portal Import Options

Field	Description
Include Services Data	Select to import portal-related data stored in the WebCenter Portal database for activity streams, events, feedback, lists, links, message boards, people connections, profiles, polls, and surveys. Deselect this option if you do not want to import any portal-related data. When you choose this option, data on the target (if any) is preserved. For example, when moving a portal from a test environment to a stage or production environment where test data is not required. Note: The import process does not move externally stored data such as mail, discussions, announcements, and so on.

7. Click **Import**.

   - If you try to import portals that exist in the target WebCenter Portal application, the **Confirm Replace** dialog displays. You must confirm whether you want to overwrite the existing portals.

     To delete existing portals and replace them with imported versions, answer **Yes**. Answer **No** to cancel the import process.

   - If the import process detects a conflict between the portals you are trying to import and those which exist on the target, a message displays to help you resolve the issue. For example, conflict messages display if a portal on the target application has the same name but a different GUID to a portal you are trying to import. In this instance you could change the name of the source portal and create a new export archive, or rename the conflicting portal in the target application and import the same archive.

   - If the portal archive exceeds the maximum upload size for files (2 GB by default) you cannot import the portals. Oracle recommends that you use the `importWebCenterPortals` WLST command to import any portal archive that exceeds the current upload size. For details, see Section 40.1.2.4.3, "Importing a Portal from an Archive Using WLST." If necessary, you can increase the upload setting. For details, see Section 9.12, "Changing the Maximum File Upload Size."

     Notes:

     - If you are working with legacy portals with assets that store artifacts in MDS, Oracle recommends that you relocate all dependent artifacts from MDS to your content server. If you choose not to move artifacts stored in MDS and do not include MDS content within the asset archive, you can use MDS WLST commands `exportMetadata`/`importMetadata` to move the MDS content another time. For example:

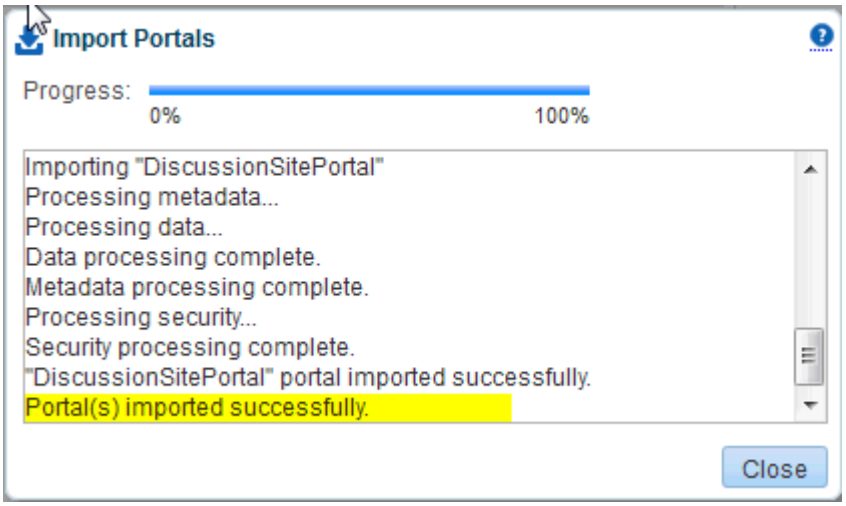
```
exportMetadata(application='webcenter', server='WC_Spaces',
 toLocation='/tmp/content',
 docs='/oracle/webcenter/siteresources/scopedMD/shared/**')

importMetadata(application='webcenter', server='WC_Spaces',
 fromLocation='/tmp/content',
 docs='/oracle/webcenter/siteresources/scopedMD/shared/**')
```

An information message displays.

8. Click **Yes** to confirm that you want to import the portals.

   An information message displays when all portals import successfully (Figure 40–10).

Figure 40–10 Portal Import Successful



9. Click **Close** to dismiss the Import Portals window.

Typically, some additional work is required before new portals are ready for general use so initially, all newly imported portals are *offline*. For example, you may want to:

- Migrate data associated with back-end components. For details, see Section 40.8.1, "Migrating Back-end Components for Individual Portals."
- Add or invite members.
- Enable or disable tools and services.

Once portal content and membership details are finalized you can bring the portal online, see Section 46.12, "Bringing Any Portal Back Online."

40.1.2.4.3 Importing a Portal from an Archive Using WLST

Use the WLST command `importWebCenterPortals` to import one or more archived portals into WebCenter Portal:

```
importWebCenterPortals(appName, fileName, names, [parentPortal,
 importCustomizations, importPortalContent, importSecurity, importData,
 importActivities, overwrite, savePortals, forceOffline, importLog])
```

When you import portals using WLST, you do not have to import everything inside the archive. If the archive contains multiple portals you can specify only those portals that you want to import. You can also specify how much information is imported along with the portals. For example you can choose whether or not to import the portal's content folder, customizations, security information, or data. These options are useful as in some circumstances, such as moving a portal from a test environment to a stage or production environment, test-related data/content is not always required.

The options that you set depends on your specific requirements. For command syntax, see the "importWebCenterPortals" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Here are a few examples:

Example 1 - Importing two portals on the target for the first time

This example imports two portals named `Sales` and `Finance`, plus all content, data, security, and customizations, and also specifies a name and location for the import log file:

```
importWebCenterPortals(appName='webcenter', fileName='MyPortalExport.par',
 names='Sales,Finance', importLog='/myimportlogs/myPortal_import.log')
```

Example 2 - Importing a portal that exists on the target

This example backs up a portal named `myExistingPortal` on the target and then overwrites the target portal with the archived version (excluding all possible data):

```
importWebCenterPortals(appName='webcenter', fileName='MyPortalExport.par',
 names='myExistingPortal', importCustomizations=0, importPortalContent=0,
 importSecurity=0,importData=0, importActivities=0, overwrite=1, savePortals=1)
```

40.1.2.5 Viewing and Extracting Portal Archives

Use the WLST command `listWebCenterPortalArchive` to view the content of a portal archive (`.par` file). You can also extract the portal archive content to a location of your choice, if required. For command syntax, see the "listWebCenterPortalArchive" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

40.1.3 Deploying Portal Hierarchies

In WebCenter Portal, a portal hierarchy comprises a parent portal with one or more subportals (as shown in Figure 40–11). The deployment process for portal hierarchies is the same as for a single portal but there are a few guidelines and restrictions due to the dependency between portals in a hierarchy. For details see:

- Guidelines for Exporting Portal Hierarchies
- Guidelines for Importing Portal Hierarchies

Guidelines for Exporting Portal Hierarchies

When you export a portal hierarchy, for deployment to another server or for backup purposes, only specify that you want to export the *parent portal* and the whole hierarchy is exported to a single portal data file (PDR) named (`<parentportalname>-<datetimestamp>.pdr`).

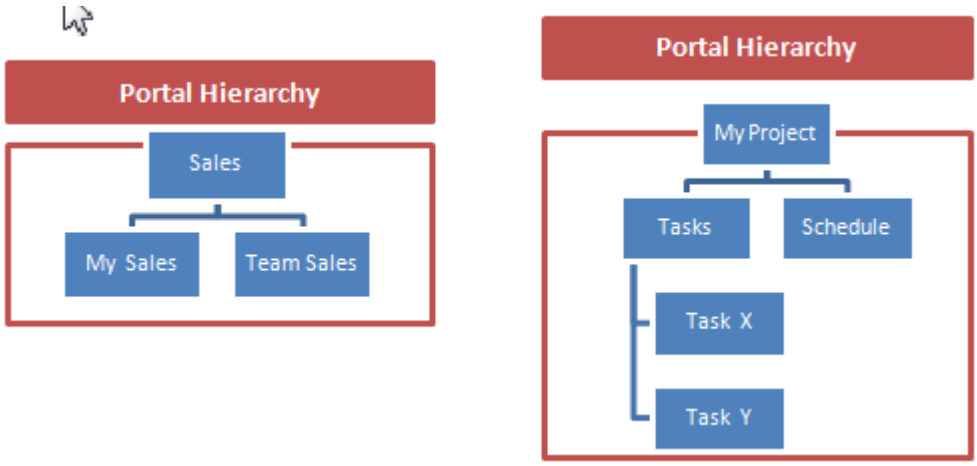
> **Important:** Due to various dependencies between portals in a hierarchy, Oracle recommends that you *always* export and import the complete portal hierarchy.

For example, to export portals in the hierarchies shown in Figure 40–11, specify the following:

```
exportWebCenterPortals(appName='webcenter', fileName='myPortalHierarchies.par',
 names='Sales,MyProject')
```

While "Tasks" is a parent portal for "Task X" and "Task Y", it is not the highest portal in the hierarchy. If you try to export only the "Tasks" hierarchy, information inherited from "My Project" will be missing from the export archive.

Figure 40–11 Deploying Portal Hierarchies



Guidelines for Importing Portal Hierarchies

- Selective import or restoration of child portals in an archive is not allowed. Always export the whole hierarchy and import the whole hierarchy on the target.
- Do not change the parent when you import a hierarchical portal on the target.

For example, to import all portals in the hierarchies shown in Figure 40–11, specify:

```
importWebCenterPortals(appName='webcenter', fileName='myPortalHierarchies.par',
 names='Sales,MyProject', forceOffline=1)
```

40.2 Deploying Portal Templates

Administrators can export portal templates from WebCenter Portal and deploy them on another portal server. Out-of-the-box templates cannot be exported.

While export and import utilities are primarily used to move information between WebCenter Portal instances, the portal template export feature is also useful as a backup service, and for sharing and exchanging templates with others.

Portal templates can contain pages, documents, discussions, lists, and security information such as custom roles and member details. When you export a portal template all this information is packaged in a portal data file (PDR). The PDR file contains a metadata archive (MAR file) and a single XML file containing security policy information for the template.

As all the template data is included in the portal template archive, you do not need to manually migrate any template data to the target when you deploy a portal template to another WebCenter Portal Server.

Portal templates that use document services (files, folders, wikis, blogs) automatically own a content folder on WebCenter Portal's back-end content repository. When you use Portal Builder administration to export portal templates, the content stored in this folder is automatically included in the portal template archive (`.pdr`) for easy deployment to another target server. The folder is added to a `.zip` file located at:

`transport.mar\oracle\webcenter\lifecycle\importexport\data\oracle-webcenter-doclib\docsexport.zip`.

If you export the portal template using the WLST command `exportWebCenterPortalTemplates` the content folder is optional.

> **Note:** Portal template archives **do not** include web content/pages referenced by the portal template that is stored at any other location, for example, information displayed through Content Presenter that is not stored in the portal template's content folder. Only the folder assigned to the portal template on WebCenter Portal's back-end content repository is included with the portal template archive.

This section includes the following topics:

- Section 40.2.1, "Exporting Portal Templates"
- Section 40.2.2, "Importing Portal Templates"

40.2.1 Exporting Portal Templates

Administrators can use the WLST command `exportWebCenterPortalTemplates` to export one or more portal templates to an archive. Alternatively, administrators and application specialists can use Portal Builder administration to export portal templates to an archive.

This section includes the following topics:

- Section 40.2.1.1, "Exporting Portal Templates to an Archive Using WebCenter Portal"
- Section 40.2.1.2, "Exporting Portal Templates to an Archive Using WLST"

40.2.1.1 Exporting Portal Templates to an Archive Using WebCenter Portal

Application specialists (and other users with the `Portal Templates-Manage All` permission) can export portal templates from WebCenter Portal. For more information, see the "Exporting Portal Templates" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

> **Note:** You cannot export portals and portal templates into a single archive. Exporting portals is a separate process. For details see Section 40.1.2.3.2, "Exporting Online Portals to an Archive Using Portal Builder Administration."

40.2.1.2 Exporting Portal Templates to an Archive Using WLST

Use the WLST command `exportWebCenterPortalTemplates` to export one or more portal templates to an archive (`.par` file). When you create a portal template archive using WLST you can choose whether or not to include the portal's content folder in the archive:

```
exportWebCenterPortalTemplates(appName, fileName, [names,
 exportPortalTemplateContent, server, applicationVersion])
```

The options that you set depends on your specific archive requirements. For command syntax, see the "exportWebCenterPortalTemplates" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Here are a few examples:

Example 1 - Exporting two portal templates

This example exports two templates named `SalesTargetTemplate` and `NewProjectTemplate`, plus their associated content folders:

```
exportWebCenterPortalTemplates(appName='webcenter',
 fileName='MyTemplateExport.par', names='SalesTargetTemplate,NewProjectTemplate',
 exportPortalTemplateContent=1)
```

Example 2 - Exporting a single portal template without its content folder

This example exports the `New Hire` template. Documents are not enabled in this template so the template does not have a content folder:

```
exportWebCenterPortals(appName='webcenter', fileName='MyTemplateExport.par',
 names='NewHire')
```

40.2.2 Importing Portal Templates

Administrators can use the WLST command `importWebCenterPortals` to deploy one or more portal templates on another WebCenter Portal Server. Alternatively, administrators and application specialists can use Portal Builder administration to import portal templates from an archive.

On import, *all* portal templates included in the archive are re-created on the target application. If a portal template exists on the target, then it is deleted and replaced. If a portal template does not exist, then it is created.

Newly imported portal templates are not immediately available for general use. You must publish newly imported templates to make them available to everyone. See the "Publishing or Hiding Portal Templates" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

This section includes the following topics:

- Section 40.2.2.1, "Importing Portal Templates from an Archive Using WebCenter Portal"
- Section 40.2.2.2, "Importing Portal Templates from an Archive Using WLST"

40.2.2.1 Importing Portal Templates from an Archive Using WebCenter Portal

Application specialists (and other users with `Portal Templates-Manage All` permission) can import portal templates into WebCenter Portal. For more information, see the "Importing Portal Templates" in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

40.2.2.2 Importing Portal Templates from an Archive Using WLST

Use the WLST command `importWebCenterPortals` to import one or more portal templates from an archive (`.par` file). When you import a portal template archive using WLST you can choose whether or not to import template content folders:

```
importWebCenterPortals(appName, fileName, names, [importPortalContent],
[overwrite], [savePortals], [importLog])
```

The options that you set depends on your specific archive requirements. For command syntax, see the "importWebCenterPortals" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Here are a few examples:

Example 1 - Importing a new portal template without content

The following example imports the `New Hire` portal template archived in `myPortalTemplateExport.par` and specifies a name and location for the import log file. Documents are not enabled in this template so the template does not have a content folder.

```
importWebCenterPortals(appName='webcenter', fileName='myPortalTemplateExport.par',
 names='NewHire', importLog='newHireTemplate_import.log')
```

Example 2 - Imports two existing portal template with content:

This example backs up portal templates named `SalesTargetTemplate` and `NewProjectTemplate` on the target, and then overwrites the existing templates and their content folders with information in `myPortalTemplateExport.par`:

```
importWebCenterPortals(appName='webcenter', fileName='myPortalTemplateExport.par',
 names='SalesTargetTemplate,NewProjectTemplate', importPortalContent=1,
 overwrite=1, savePortals=1, importLog='myPortalTemplate_import.log')
```

40.3 Deploying Assets

Authorized users can download assets, such as skins and page templates, while WebCenter Portal is running, edit and extend them in tools such as Oracle JDeveloper, and then deploy them back to WebCenter Portal. Users who want to share assets or migrate assets to other WebCenter Portal instances can use the download/upload feature too.

WebCenter Portal users can download and upload the following assets through Portal Builder and administrators can perform the same tasks using WLST commands:

- Page templates
- Navigations
- Resource catalogs
- Skins
- Page styles
- Content presenter display templates
- Task flow styles
- Task flows
- Data controls

> **Note:** While you cannot upload or download individual pagelets, all assets (including pagelets) are included when you migrate individual portals or an entire WebCenter Portal instance.

When you download (or export) a WebCenter Portal asset, the asset details are saved to an export archive (`.ear` file). You can save the export archive to your local file system or a remote server file system using a filename of your choice.

External Asset Dependencies

No artifacts used or referenced by assets, such as icons, images, and data controls, are included in the archive. When you upload an asset to another WebCenter Portal instance you are must manage and move dependent asset artifacts manually. Oracle recommends that you use a folder structure on your content server specifically for asset artifacts so that content is easy to identify and move, if required.

Note: If you are managing legacy assets that store artifacts in MDS, Oracle recommends that you relocate dependent artifacts to your content server. However, if you do need to move artifacts stored in MDS, you can use the MDS WLST commands `exportMetadata/importMetadata`. For example:

```
exportMetadata(application='webcenter', server='WC_Spaces',
 toLocation='/tmp/content',
 docs='/oracle/webcenter/siteresources/scopedMD/shared/**')

importMetadata(application='webcenter', server='WC_Spaces',
 fromLocation='/tmp/content',
 docs='/oracle/webcenter/siteresources/scopedMD/shared/**')
```

This section includes the following topics:

- Section 40.3.1, "Exporting Assets to an Archive"
- Section 40.3.2, "Importing Assets from an Archive"

40.3.1 Exporting Assets to an Archive

This section describes the various ways you can create an asset archive. It includes the following topics:

- Section 40.3.1.1, "Exporting Assets to an Archive from Portal Builder"
- Section 40.3.1.2, "Exporting Assets to an Archive using WLST"
- Section 40.3.1.3, "Exporting Assets to an Archive from JDeveloper"

See also, Section 40.3.2.1, "About Permissions Required to Import (or Export) Assets."

40.3.1.1 Exporting Assets to an Archive from Portal Builder

Administrators, application specialists, and portal moderators can export assets from Portal Builder. For details, see the "Downloading an Asset" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

40.3.1.2 Exporting Assets to an Archive using WLST

Administrators can use the WLST command `exportWebCenterResource` to export a single asset from WebCenter Portal:

```
exportWebCenterResource(appName, fileName, resourceType, [resourceGUID,
 resourceName, spaceName, exportContentDirectory,server, applicationVersion])
```

The options that you set depends on the asset you want to export. For command syntax, see the "exportWebCenterResource" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Here are a few examples:

Example 1 - Exporting a page template belonging to the "Sales" portal

The following example exports a page template from the `Sales` portal to a file named `mySalesPageTemplateExport.ear`:

```
exportWebCenterResource(appName='webcenter',
 fileName='mySalesPageTemplateExport.ear', resourceType='pageTemplate',
 resourceGUID='gsr47d9a5ac_7398_439a_97d2_8b54ce905f7e, spaceName='SalesPortal')
```

Example 2 - Exporting a shared portal skin identified by GUID

The following example exports a shared portal skin to a file named `mySharedSkinExport.ear`:

```
exportWebCenterResource(appName='webcenter', fileName='mySharedSkinExport.ear',
 resourceType='skin', resourceGUID='gsr5a8c2fcc_bc7f_4cba_9254_36df58d66e60)
```

Example 3 - Exporting a shared portal skin identified by name

The following example exports the same shared portal skin but specifies the skin's display name rather than the GUID:

```
exportWebCenterResource(appName='webcenter', fileName='mySharedSkinExport.ear',
 resourceType='skin', resourceName='MyCompanySkin)
```

40.3.1.3 Exporting Assets to an Archive from JDeveloper

Developers can export assets that they create or extend in JDeveloper to an archive (`.ear` file) for others to deploy. For more information, see the "Exporting WebCenter Portal Assets to an Archive" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

40.3.2 Importing Assets from an Archive

You can only import an asset previously saved to a WebCenter Portal export archive (`.ear` file). For details, see Section 40.3.1, "Exporting Assets to an Archive."

On import:

- *Existing assets* are overwritten, that is, assets with the same internal ID.
- *Portal assets* are always imported back into the same portal. You cannot import a resource into a different portal.

This section describes the various ways you can import an asset to WebCenter Portal from an archive. It includes the following topics:

- Section 40.3.2.1, "About Permissions Required to Import (or Export) Assets"
- Section 40.3.2.2, "Importing Assets from an Archive using Portal Builder"
- Section 40.3.2.3, "Importing Assets from an Archive using WLST"

40.3.2.1 About Permissions Required to Import (or Export) Assets

Table 40–6 describes the roles/permission required to import (or export) assets using the Portal Builder administration.

> **Note:** If you want to import (or export) assets using WLST, you must also have the WebLogic Server `Monitor` role (or higher).

Table 40–6 Permissions Required to Import (or Export) Assets Using Portal Builder

Asset	Required WebCenter Portal Role or Permission	Description
Shared asset	▪ `Administrator`	▪ This role includes the required permissions for importing and exporting shared assets (`Create, Edit, Delete Assets` and `Manage Configuration`). See also, Section 49.6, "Managing Application Roles and Permissions."
	OR	
Shared asset	▪ `Create, Edit, Delete <resourcetype>`	▪ This permission enables you to create and manage shared assets for WebCenter Portal.
	▪ `Manage Configuration`	▪ This application-level permission (`Manage Configuration`) gives you access to Portal Builder administration pages.
Portal asset	▪ `Moderator`	▪ This role includes the required permissions (`Create, Edit, Delete Assets` and `Manage Configuration`). See also, the "Managing Roles and Permissions for a Portal" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.
	OR	
Portal asset	▪ `Create, Edit, Delete Resources` (standard) OR `Create, Edit, Delete <resourcetype>` (advanced)	▪ These permissions enable you to create and manage assets for a particular portal. Either standard or advanced permissions will apply, depending on the portal.
	▪ `Manage Configuration`	▪ This portal-level permission (`Manage Configuration`) gives you access to the asset administration page for a particular portal.

40.3.2.2 Importing Assets from an Archive using Portal Builder

Administrators, application specialists, and portal moderators can import assets from Portal Builder. For details, see the "Uploading an Asset" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*

40.3.2.3 Importing Assets from an Archive using WLST

Administrators can use the WLST command `importWebCenterResource` to deploy a single asset to WebCenter Portal.

```
importWebCenterResource(appName, fileName, [resourceType, spaceName,
 overwriteContentDirectory, server, applicationVersion])
```

The options that you set depends on the asset you want to deploy. For command syntax, see the "importWebCenterResource" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Here are a few examples:

Example 1 - Deploying a page template to the "Sales" portal

The following example imports a page template archived in `mySalesPageTemplateExport.ear` in to the `Sales` portal:

```
importWebCenterResource(appName='webcenter',
 fileName='mySalesPageTemplateExport.ear', resourceType='pageTemplate',
 spaceName='SalesPortal')
```

Example 2 - Deploying a shared portal skin

The following example imports a shared portal skin archived in `mySharedSkinExport.ear`:

```
importWebCenterResource(appName='webcenter', fileName='mySharedSkinExport.ear',
 resourceType='skin')
```

40.4 Deploying Devices and Device Groups

Administrators can export device groups and devices to a file (`.ear` file), and then import (deploy) them to another WebCenter Portal instance. For example, if you want to move devices or device groups developed on stage to a production server or share your devices and device groups with another WebCenter Portal installation.

> **Note:** You cannot export or import out-of-the-box device groups or devices. You can only export and import device groups or devices that you and other administrators create or copy

This section includes the following topics:

- Section 40.4.1, "Exporting Devices and Device Groups to an Archive"
- Section 40.4.2, "Importing Devices and Device Groups from an Archive"

40.4.1 Exporting Devices and Device Groups to an Archive

This section includes the following topics:

- Section 40.4.1.1, "Exporting Devices and Device Groups Using Portal Builder"
- Section 40.4.1.2, "Exporting Devices and Device Groups Using WLST"

40.4.1.1 Exporting Devices and Device Groups Using Portal Builder

Administrators can export one or more devices and device groups to a file (`.ear` file) from Portal Builder Administration. For details, see Section 53.4, "Managing Device and Device Group Life Cycles."

40.4.1.2 Exporting Devices and Device Groups Using WLST

Administrators can use the WLST command `exportWebCenterResource` to export a single device or device group from WebCenter Portal to a file (`.ear` file):

```
exportWebCenterResource(appName, fileName, resourceType, [resourceName, server,
 applicationVersion])
```

For command syntax, see the "exportWebCenterResource" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Here are a few examples:

Example 1 - Exporting a device group

The following example exports a device group named "MyMobileDeviceGroup" from WebCenter Portal:

```
exportWebCenterResource(appName='webcenter', fileName='myDeviceGroupExport.ear',
 resourceType='deviceGroup', resourceName='MyMobileDeviceGroup)
```

Example 2 - Exporting a device

The following example exports a device named "MyMobileDevice" from WebCenter Portal:

```
exportWebCenterResource(appName='webcenter', fileName='myDeviceExport.ear',
 resourceType='device', resourceName='MyMobileDevice)
```

40.4.2 Importing Devices and Device Groups from an Archive

This section includes the following topics:

- Section 40.4.2.1, "Importing Devices and Device Groups Using Portal Builder"
- Section 40.4.2.2, "Importing Devices and Device Groups Using WLST"

40.4.2.1 Importing Devices and Device Groups Using Portal Builder

Administrators can import one or more devices and device groups from a file (.ear file) using Portal Builder Administration. For details, see Section 53.4, "Managing Device and Device Group Life Cycles."

40.4.2.2 Importing Devices and Device Groups Using WLST

Administrators can use the WLST command `importWebCenterResource` to deploy one or more devices or device groups to WebCenter Portal from a file (`.ear` file).

```
importWebCenterResource(appName, fileName, [resourceType, server,
 applicationVersion])
```

For command syntax, see the "importWebCenterResource" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Here are a few examples:

Example 1 - Deploying a device group

The following example imports a device group exported to `myDeviceGroupExport.ear`:

```
importWebCenterResource(appName='webcenter', fileName='myDeviceGroupExport.ear',
 resourceType='deviceGroup')
```

Example 2 - Deploying a device

The following example imports a device archived in `myDeviceExport.ear`:

```
importWebCenterResource(appName='webcenter', fileName='myDeviceExport.ear',
 resourceType='device')
```

40.5 Deploying Custom Shared Library Extensions

Developers can use JDeveloper to build custom ADF library components for portals, such as managed beans, task flows, and data controls, and deploy them as shared library extensions to the portal server.

See also, the "Developing Components for WebCenter Portal Using JDeveloper" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

40.6 Moving Connections Details from Staging to Production

Administrators can use the WLST commands `exportWebCenterPortalConnections` and `importWebCenterPortalConnections` to migrate connections details from one WebCenter Portal installation to another. These commands are useful if you import or restore a portal and connections used in the source server, such as portlet producer connections and web service connections, do not exist on the target server.

For more information on the types of connections you can migrate, see Section 40.1.2.1.3, "Understanding Connection Property Files."

This section includes the following topics:

- Section 40.6.1, "Exporting WebCenter Portal Connections Details to a File"
- Section 40.6.2, "Importing New WebCenter Portal Connections from a File"

40.6.1 Exporting WebCenter Portal Connections Details to a File

If you have WebLogic Server `Operator` role (or higher) you can use the WLST command `exportWebCenterPortalConnections` to export connection information currently configured for a particular WebCenter Portal installation to a file:

```
exportWebCenterPortalConnections(appName, fileName, [connectionType,
 connectionName, logFile, server, applicationVersion])
```

> **Note:** You cannot export connections for a specific portal. Connections are shared across all the portals.

The options that you set depends on the connection information you want to export. For command syntax, see the "exportWebCenterPortalConnections" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Here are a few examples:

Example 1 - Deploying all WSRP producer and external application connections to a file

The following example only exports WSRP producer and external application connections to a file named `myconnection.properties`:

```
exportWebCenterPortalConnections(appName='webcenter',
 fileName='/myConnections/myconnection.properties',
 connectionType='wsrpProducerConnection,externalAppConnection')
```

Example 2 - Deploying specific WSRP producer connections to a file

The following example exports connection configuration information for two WSRP producer connections named `MyWSRP1` and `MyWRSP2`:

```
exportWebCenterPortalConnections(appName='webcenter',
 fileName='/myConnections/connection.properties',
 connectionType='wsrpProducerConnection', connectionName='MyWSRP1,MyWSRP2')
```

40.6.2 Importing New WebCenter Portal Connections from a File

If you have WebLogic Server `Operator` role (or higher) you can use the WLST command `importWebCenterPortalConnections` to deploy connection information exported from one WebCenter Portal installation to another.

```
importWebCenterPortalConnections(appName, fileName, [promptForPassword, logFile,
 server, applicationVersion])
```

Only new connections are imported on the target. Connections that already exist on the target are ignored. The source connection information must be exported using the WLST command `exportWebCenterPortalConnections`. To find out how, see Section 40.6.1, "Exporting WebCenter Portal Connections Details to a File."

If required, you can edit the file that contains the connection information *before* you deploy the connection information on the target. See also Section 40.1.2.1.3, "Understanding Connection Property Files."

Example 1 - Importing connections from a file

The following example imports connections defined in a file named `myconnection.properties` located in `/myConnections`. Detailed information about the import connection operation is also logged to `importConnection.log`:

```
importWebCenterPortalConnections(appName='webcenter',
 fileName='/myConnections/myconnection.properties',logFile='importConnection.log')
```

Example 2 - Importing connections that require credentials

The following example imports connections defined in a file named `myconnection.properties` located in `/myConnections` and prompts you for credentials if required:

```
importWebCenterPortalConnections(appName='webcenter',
 fileName='/myConnections/myconnection.properties', promptForPassword=1)
```

For command syntax, see the "importWebCenterPortalConnections" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

40.7 Moving Portals from Staging to Production

If you are using a staging environment to develop and test your portals before moving them to your production server, Oracle recommends that you *always make changes in stage first* and move changes to production to keep both environments in sync. The steps for moving the portals are exactly the same as those described in, Section 40.9, "Managing Portals in Production."

40.8 Moving External Portal Data from Staging to Production

This section includes the following:

- Section 40.8.1, "Migrating Back-end Components for Individual Portals"
- Section 40.8.2, "Migrating Back-end Components for an Entire Portal Server"

40.8.1 Migrating Back-end Components for Individual Portals

After you move/migrate one or more portals to another server, you can (optionally) migrate portal data that is stored by various back-end components.

Discussions:

- Section 40.8.1.1, "Exporting Portal Discussions to an Archive"
- Section 40.8.1.2, "Importing Portal Discussions from an Archive"

Content:

- Section 40.8.1.3, "Exporting Content for a Portal"
- Section 40.8.1.4, "Importing Content for a Portal"

Pagelets:

- Section 22.5.1, "Exporting and Importing Pagelet Producer Resources"
- Section 22.5.2, "Exporting and Importing Pagelet Producer Metadata Using WLST"

After importing one or more portals, consider initiating an Oracle Secure Enterprise Search crawl to index the newly imported data.

40.8.1.1 Exporting Portal Discussions to an Archive

Use the discussions server's Admin Console to export discussions associated with a particular portal.

Portal discussions are exported to an `.xml` file, and saved to a `.zip` file in the `DOMAIN_HOME/config/fmwconfig/servers/<target_server_name>/owc_discussions/data/` directory.

Where `DOMAIN_HOME` is the path to the Oracle WebLogic Server domain. For example, `MW_HOME/user_projects/domains/my_domain/config/fmwconfig/servers/WC_Collaboration/owc_discussions/data/`.

To export discussions for a portal:

1. Login to the Admin Console for the discussions server.

   You can login directly if you know the console's URL. For example: `http://example.com:8890/owc_discussions/admin`

   Alternatively, log in through WebCenter Portal as follows:

   a. Open WebCenter Portal administration.

      For details, see Chapter 47, "Exploring the Administration Page in Portal Builder Administration."

   b. Click **Portals**.

   c. Select the portal whose discussions you want to export, then select **Administer**.

   d. Click **Tools and Services**, then **Discussions**.

e. Note down the **Forum Name/Forum ID** or **Category Name/Category ID** associated with the portal.

   WebCenter Portal's discussions server generates discussion category and forum IDs sequentially. If this ID exists on the target system, the imported forum (or category) will be assigned a new, unique ID, and therefore you must reconfigure the imported portal, to point to the new ID. For details, see Step 11 below.

f. Click **Administer Forums**, and login to the Discussions Server Admin Console.

2. In the Admin Console, select the **System** menu and select **XML Import & Export** in the sidebar.
3. Select **Data Export**.
4. Set the following options (Figure 40–12):

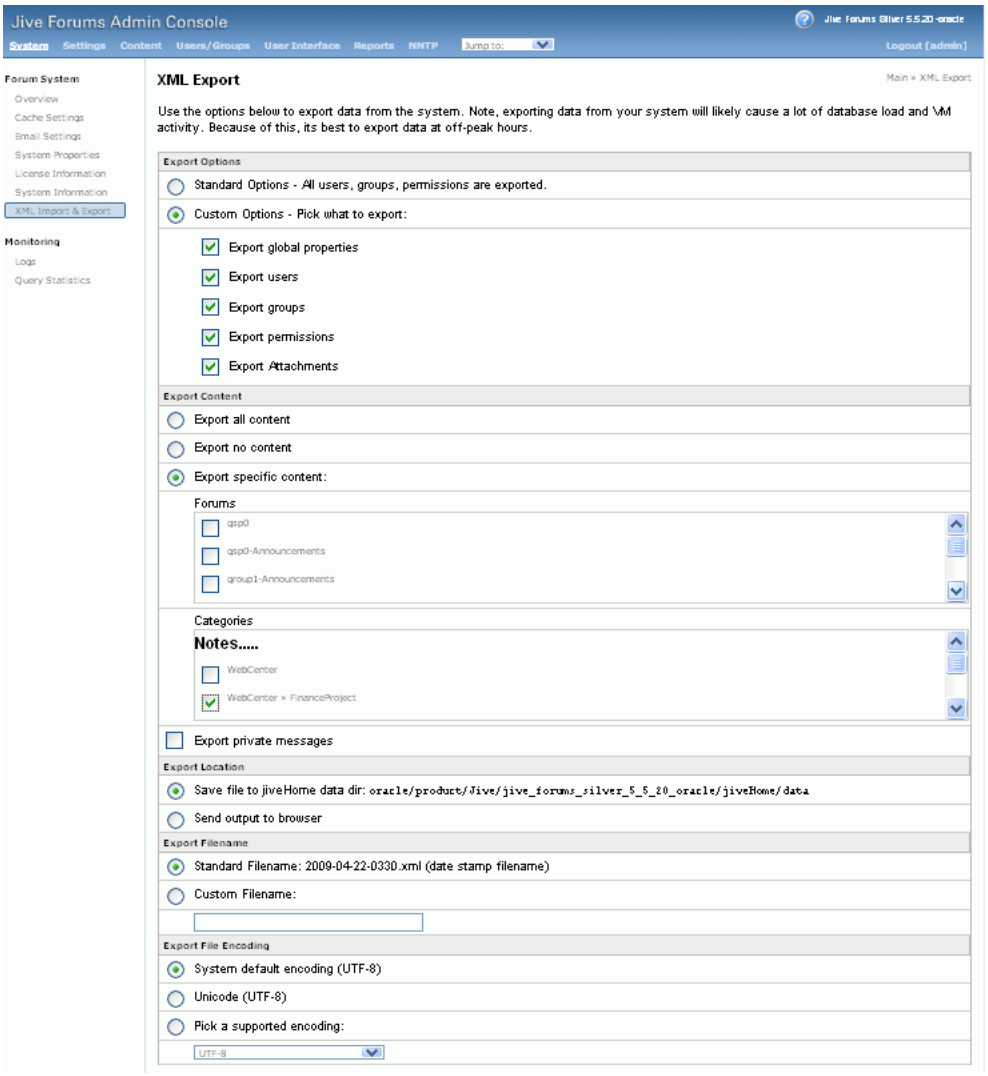
   a. **Export Options** - Select **Custom Options**, and select all the check boxes.

   b. **Export Content** - Select **Export Specific Content**, and select the name of the forum or category required.

      Note: Portals that support multiple forums use a category to store discussions. Other portal use a single forum.

   c. **Export location, Export filename, Export file encoding** - Keep the default values.

Figure 40–12 Exporting Discussions for an Individual Portal



5. Click **Start Export**.
6. Once complete, copy the `.zip` file (that contains the export `.xml` file) from the `MW_HOME/user_projects/domains/my_domain/config/fmwconfig/servers/<server_name>/owc_discussions/data` directory to same location on the target discussions server.

   For example:
   `MW_HOME/user_projects/domains/my_domain/config/fmwconfig/servers/WC_Collaboration/owc_discussions/data`

Before importing discussions on the target system, the portal you are migrating must exist on the target. See Section 40.1.2.4.2, "Importing a Portal from an Archive Using Portal Builder Administration."

40.8.1.2 Importing Portal Discussions from an Archive

Use the discussions server's Admin Console to import discussions exported from another WebCenter Portal environment.

Ensure that the associated portal exists on the target *before* you import the discussion data. See Section 40.1.2.2, "Deploying Portal Archives to a Different Server" or

Section 40.9.2, "Directly Deploying Portals From Staging to Production."

> **Note:** WebCenter Portal's discussions server generates discussion category and forum IDs sequentially. Therefore, when importing discussion data between two targets (or source to target), there is a chance that the same IDs exist on both systems. When ID clashes occur, the imported forum (or category) is assigned a new, unique ID and therefore you must reconfigure the portal to point to the new ID. See Step 11 below for details.

To import discussions for a particular portal:

1. Log into the Admin Console for the target discussions server.

   You can login directly if you know the console's URL. For example:
   `http://example.com:8890/owc_discussions/admin`

   Alternatively, log in through WebCenter Portal as follows:

   a. Open WebCenter Portal administration.

      For details, see Chapter 47, "Exploring the Administration Page in Portal Builder Administration."

   b. Click **Portals**.

   c. Select the portal for which you want to import data, and then select **Administer**.

   d. Click **Tools and Services**, then **Discussions**.

   e. Click **Administer Forums** (on the far right), and log into the Admin Console.

2. In the Admin Console, select the **System** menu and then select **XML Import & Export** in the sidebar.

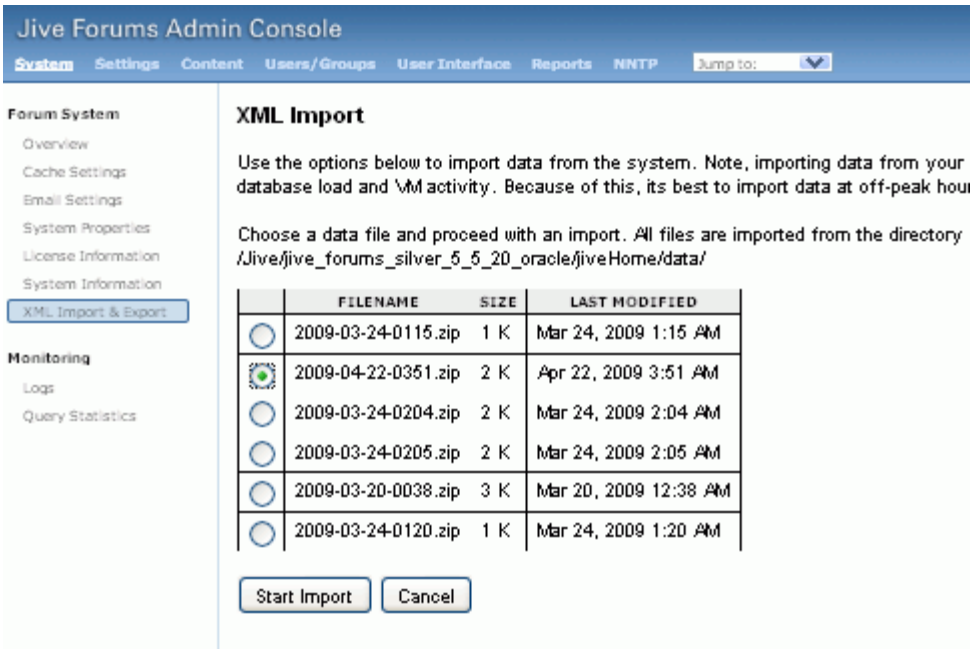
3. Select **Data Import**.

4. Select the appropriate import file from the list available (Figure 40–13).

   If the file you want is not listed, copy the export `.zip` file from the source directory `DOMAIN_HOME/config/fmwconfig/servers/<target_server_name>/owc_discussions/data/` to same location on this target. See also, Section 40.8.1.1, "Exporting Portal Discussions to an Archive."

   Where `DOMAIN_HOME` is the path to the Oracle WebLogic Server domain. For example:
   `MW_HOME/user_projects/domains/my_domain/config/fmwconfig/servers/WC_Collaboration/owc_discussions/data/`

Figure 40–13 Importing Discussions for a Portal



5. Click **Start Import**.

   On import, the discussions data is copied to the discussions server. In the next step you reassociate the portal you migrated earlier with this newly imported data.

6. Select the **Content** menu, and then select **Content Summary** in the sidebar.

   All the categories and forums in the system are listed here.

7. Select **WebCenter**, and then click the **Move** button for the newly imported forum or category.

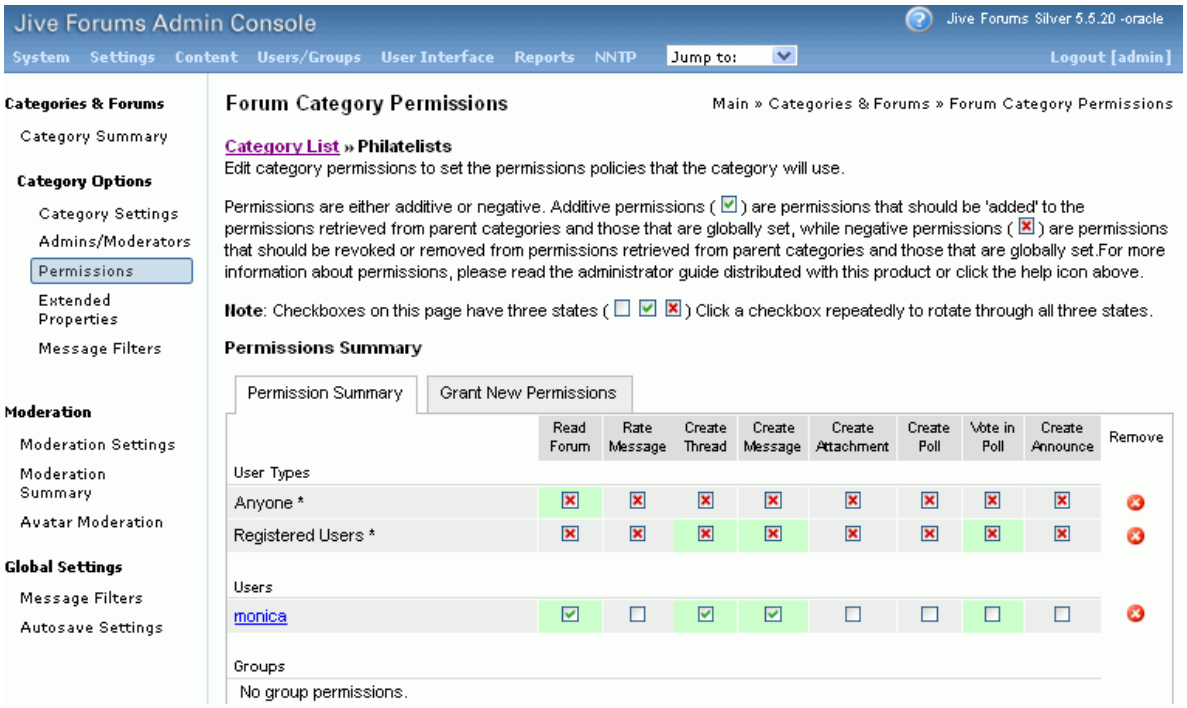
8. Select the root category for the target WebCenter Portal, and click **Move Categories**.

   The Category Summary page shows the new location.

9. Click **Permissions** in the sidebar.

10. Deselect all the permissions for the User Types: **Anyone** and **Registered Users**, and click **Save Changes** (Figure 40–14).

Figure 40–14 Editing Forum Permissions



11. In WebCenter Portal, navigate to Discussions Forum Settings for the portal to reassociate the portal with the discussion data that you just imported:

    a. Open WebCenter Portal administration.

       For details, see Chapter 47, "Exploring the Administration Page in Portal Builder Administration."

    b. Click **Portals**.

    c. Select the portal for which you want to import data, and then select **Administer**.

    d. Click **Tools and Services**, then **Discussions**.

    e. Click the **Search** icon besides Category ID or Forum ID, select the imported category (or forum) from the list, and click `Select`.

    f. Click **Save**.

40.8.1.3 Exporting Content for a Portal

Oracle recommends that documents and content associated with your portal are placed in the *portal folder*. When you export a portal to an archive, there is an option to include this portal folder in the portal archive (see Section 40.1.2.3, "Creating Portal Archives") and there is a similar option when you deploy a portal directly to another portal server (see Section 40.9.2, "Directly Deploying Portals From Staging to Production").

> **Note:** If you choose not to migrate the portal folder with the portal you can manually move the portal folder to the target using WebCenter Content Server migration tools. For details, see the "System Migration and Archiving" section in *Oracle Fusion Middleware Administering Oracle WebCenter Content*.

In addition to the portal folder, your portal may reference content in other locations, for example:

- Other folders on Content Server
- External content repositories or web locations

You cannot export content in any other folder or location along with the portal so you must ensure the target system can access the same content as the source. See also, Section 40.8.1.4, "Importing Content for a Portal."

In all cases, documents and content remain in the source content repository so if required, you can migrate any content that you require after migrating the portal to the target.

40.8.1.4 Importing Content for a Portal

When you import a portal from Portal Builder Administration, you can select "Include Portal Folder on Content Server" in the import dialog if you want to import documents with the portal. If you are using the `importWebCenterPortals` WLST command to do the import, set `importPortalContent=1`.

> **Note:** Portal archives do not always contain the portal folder. To include the portal folder you must select the "Include Portal Folder on Content Server" option on export as well. For details, see Section 40.1.2.3, "Creating Portal Archives."

Content Not Stored in the Portal Folder

Portal archives never include content that is stored outside the portal's own content folder. If your portal contains portal assets, portal pages, Content Presenter display templates, or other components that reference content outside the portal folder, you must either:

- Manually move such content to the target.
- Ensure that the target can access the same content as the source.

> **Note:** When you move a portal to a different target, Content Presenter data references are only maintained if Content Server connection names and root folder names are the same in both the source and the target.

For information about Oracle WebCenter Content Server migration tools, see the "System Migration and Archiving" section in *Oracle Fusion Middleware Administering Oracle WebCenter Content*. For other content repositories, refer to the manufacturer's product documentation.

40.8.2 Migrating Back-end Components for an Entire Portal Server

Before you move/migrate an entire portal server, you must migrate all the back-end components that are used by the application, see Section 41.5, "Migrating Entire WebCenter Portal to Another Target."

40.9 Managing Portals in Production

This section includes the following topics:

40.9.1 Understanding Portal Propagation

Administrators can propagate *metadata changes* in staging to production providing that your stage and production environments are connected and kept "in sync". Oracle strongly recommends that you *always* make changes in stage first and then push your metadata changes to production using deployment or propagation.

The propagation feature is useful for migrating changes to portal metadata (pages, assets, portlets dropped on to pages, and so on). Propagation does not require the production server to be restarted or incur any downtime because propagation only transfers changes to portal metadata. For details, see Table 40–7.

Table 40–7 Portal Changes Propagated to Production

Portal Changes Propagated	Yes / No
Portal-level customizations (metadata)	
Portal pages and system pages	Yes
Portlets	Yes
Assets	Yes
Task flows	Yes
User-level customizations (metadata)	
Portal pages	Yes
Portlets	No
Task flow instances	Yes
Portal folder content	No
Subportals	No
Portal activity/usage data (activity streams, calendar events, feedback, list, links, message boards, people connections, profiles, polls, surveys)	No
Portal security data (portal roles and permissions, member details and their role assignments)	No
External content referenced by the portal (through portal pages, portal assets, Content Presenter display templates, Site Studio, and so on...)	No
Data stored on external servers (discussions, mail, announcements, analytics, custom task flows and shared libraries)	No

Labeling During Propagation

For propagation to work, you must move deploy the portal to production using the WLST command `deployWebCenterPortal`—this command re-creates the entire stage portal on the production instance and creates a matching label on the stage portal and the production portal, for example LABEL_1. Subsequently, whenever you propagate changes from stage to production using the WLST command `propagateWebCenterPortal` command, the portal's label increments by 1 in both the stage and production environment, as summarized in this table:

Portal Change in Stage	Action	WLST Command	Label
First time only	Deploy the stage portal to production	`deployWebCenterPortal`	LABEL_1
Subsequent change	Propagate metadata changes in the stage portal to production	`propagateWebCenterPortal`	LABEL_2
Subsequent change	Propagate more metadata changes	"	LABEL_3
Subsequent change...	"	"	LABLE_n+1

See also, Appendix E, "Labeling During WebCenter Portal Lifecycle."

40.9.2 Directly Deploying Portals From Staging to Production

Administrators can use the WLST command `deployWebCenterPortal` to deploy a single, online portal directly to another target server. If the portal you specify has one or more subportals, the dependent portals are deployed too. You *must* use this method to deploy your portal if you want to propagate portal metadata changes in the source, such as page or asset updates, to the target.

Before deploying a portal you must complete a few prerequisite tasks. The overall process is as follows:

- Step 1: Complete prerequisites for direct portal deployment
- Step 2: Run deployWebCenterPortal in the source environment
- Step 3: Verify newly deployed portal in the target environment

Step 1: Complete prerequisites for direct portal deployment

Before running the WLST command `deployWebCenterPortal`, complete the following:

1. Verify that the name of the managed server on which WebCenter Portal is deployed is the same in both the source and target environments. For example, WC_Spaces.

   You can only run `deployWebCenterPortal` if the managed server names match. If the managed server name is different, use portal archive deployment instead. as described in Section 40.1.2.3, "Creating Portal Archives."

2. Verify that you have at least the WebLogic Server `Monitor` role and the WebCenter Portal permission `Portals - Manage All`.

3. Ensure a connection exists between the source and target WebCenter Portal.

If a connection does not exist yet, use the WLST command `adf_createHttpURLConnection` to create a connection to the target from the source environment.

For example, in the source environment run:

```
adf_createHttpURLConnection(appName='webcenter',name='MyWebCenterPortalTarget',
 url='http://example.com:7777', user='myuser', password='mypassword',
 realm='ProductionRealm')
```

See also,Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

4. (Optional) Move externally stored portal data to the target environment.

   For details, see Section 40.8.1, "Migrating Back-end Components for Individual Portals."

5. (Optional) Ensure that all the connections used by the portal (and portal components) are configured on target.

   The `deployWebCenterPortal` command does not, for example, carry connections configured for web service data controls. Connections that do not exist on the target must be migrated separately. For details, see Section 40.6, "Moving Connections Details from Staging to Production."

Step 2: Run deployWebCenterPortal in the source environment

In the source WebCenter Portal:

1. Start the WLST tool from your source WebCenter Portal Oracle home directory, and connect to the Administration Server for WebCenter Portal.

   For details, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

2. Run the WLST command `deployWebCenterPortal` to deploy the portal on the target server.

   ```
   deployWebCenterPortal(appName, portalName, targetConnectionName
    [deployCustomizations, deployContent, deploySecurity, deployData,
    deployActivities, overwrite, savePortal, deployLog, server,
    applicationVersion])
   ```

   For detailed command syntax and descriptions, see the "deployWebCenterPortal" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

   The options that you set depends on your specific deployment requirements. Here are two common examples:

   Example 1 - Deploying a portal on the target for the first time

   This example deploys a new portal named `myPortal`, plus all its associated content, data, security, and customizations, and also specifies a name and location for the deploy log file:

   ```
   deployWebCenterPortal(appName='webcenter',portalName='myPortal',
   targetConnectionName='MyWebCenterPortalTarget',deployCustomizations=1,
   deployPortalContent=1,deploySecurity=1,deployData=1,deployActivities=1,
   deployLog='/mydeploylogs/myPortal_deploy.log')
   ```

Note: Always set `deploySecurity=1` when importing a brand new portal as you cannot import a new portal without a security policy.

Example 2 - Redeploying a portal that exists on the target

This example backs up a portal named `myExistingPortal` on the target and then overwrites the target portal (`overwrite=1`) with the source portal. The content, data, security, and customizations associated with the target portal is preserved:

```
deployWebCenterPortal(appName='webcenter',portalName='myExistingPortal',
targetConnectionName='MyWebCenterPortalTarget',deployCustomizations=0,
deployPortalContent=0,deploySecurity=0,deployData=0,overwrite=1,savePortal=1)
```

3. Examine the deployment log file.

   This file is either available at the location you specified (`deployLog`) or in a file named `PortalDeploy_<timestamp>.log` in your temporary directory.

Step 3: Verify newly deployed portal in the target environment

In the target WebCenter Portal:

1. Log in to the target WebCenter Portal.
2. Navigate to the new portal deployment.
3. Verify that the portal works as expected.

40.9.3 Propagating Portal Changes in Staging to Production

Direct portal propagation is only possible if a connection exists between the source and target environments and the portal was previously deployed directly to the target using `deployWebCenterPortals` WLST command. See Section 40.9.2, "Directly Deploying Portals From Staging to Production."

To propagate metadata changes from staging to production:

1. Run the WLST command `adf_createHttpURLConnection` to create a connection to the production server:

   ```
   adf_createHttpURLConnection(appName, name, url, user, password, realm)
   ```

2. Run the WLST command `propagateWebCenterPortal` to propagate metadata for the portal.

   ```
   propagateWebCenterPortal(appName, portalName, targetConnectionName,
    [savePortal, propagateLog, server, applicationVersion])
   ```

   The options that you set depends on your specific requirements. For command syntax, see the "propagateWebCenterPortal" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

   For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Here are some examples:

Example 1 - Propagating portal metadata changes

The following commands create a connection to the production server (`MyProductionConnection`) and then propagates metadata changes for a portal named `myPortal` to the target server:

```
adf_createHttpURLConnection(appName='webcenter', name='MyProductionConnection',
 url='http://example.com:7777', user='myuser',  password='mypassword',
 realm='ProductionRealm')

propagateWebCenterPortal(appName='webcenter', portalName='myPortal',
 targetConnectionName='MyProductionConnection')
```

Example 2 - Backing up the target portal before propagating portal metadata changes

The following example backs up `myPortal` on the target, propagates metadata changes for a portal named `myPortal`, and also specifies a name and location for the propagation log file:

```
propagateWebCenterPortal(appName='webcenter', portalName='myPortal',
 targetConnectionName='MyProductionConnection', savePortal=1
 propagateLog='/mypropagationlogs/myPortal_propagation.log')
```

40.10 Restrictions

Portal deployment and portal propagation operations are only supported across two WebLogic Server instances or two IBM WebSphere instance. You cannot deploy or propagate portals from IBM WebSphere to Oracle WebLogic Server or vice versa.

41

Managing WebCenter Portal Backup, Recovery, and Cloning

This chapter describes techniques and tools for backing up and restoring WebCenter Portal installations.

This chapter includes the following topics:

Permissions: The content of this chapter is intended for system administrators.

For more information on which roles and permissions are required to deploy portals, templates, assets, connections, and extensions, see Section 39.6, "Permissions Required to Perform WebCenter Portal Life Cycle Operations."

See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

41.1 Understanding WebCenter Portal Back Up and Recovery

To recover data from disasters, such as the loss of database hardware, inadvertent removal of data from file or database, it is important to back up individual portals as well as the entire WebCenter Portal instance on a frequent basis. The frequency of your backups depend on how often the underlying information stored by WebCenter Portal changes in your particular environment, and how much time and amount of information could acceptably be lost. Incremental or partial backups may be applied where the data is critical to the business and must be restored due to a failure.

WebCenter Portal provides various back up options. Administrators can back up:

- **One or more portals, or portal hierarchies**

  WebCenter Portal provides export and import WLST commands for backing up and restoring individual portals. For details, see Section 41.3, "Backing Up Individual Portals" and Section 41.4, "Restoring Portals from a Backup."

- **Entire WebCenter Portal environment**

  Back up and recovery of WebCenter Portal as well as various back-end components can be managed through database export and import utilities, and various other tools. For more information, see Section 41.6, "Backing Up an Entire WebCenter Portal Installation" and Section 41.8, "Using Scripts to Back Up and Restore WebCenter Portal."

- **WebCenter Portal metadata, data, and security**

  WebCenter Portal provides export and import WLST commands and a methodology for backing up or migrating data owned by WebCenter Portal: WebCenter Portal MDS data (including shared assets, business role pages, and system page customizations), database data (WEBCENTER database schema only), and security. For details, see Section 41.5, "Migrating Entire WebCenter Portal to Another Target."

Note: This chapter only describes techniques for backing up and restoring WebCenter Portal data. For information about Oracle Fusion Middleware back up and recovery strategies, see the "Advanced Administration: Backup and Recovery" section in *Oracle Fusion Middleware Administrator's Guide*.

41.2 Comparing Back up, Recovery, and Migration Tools for WebCenter Portal

Table 41–1 compares the various tools available to back up and restore WebCenter Portal or migrate WebCenter Portal to another target.

Table 41–1 Backup, Restore, and Migration Tools for WebCenter Portal

	Backup and Restore	Backup and Restore Scripts	Migration / Backup
	Portals / Portal Templates	**Full WebCenter Portal Install**	**WebCenter Portal Only**

How to execute	WLST commands:
	`exportWebCenterPortals`
	`exportWebCenterPortalTemplates`
	`importWebCenterPortals`
		Customizable scripts based on:
		`master_script.sh`, `wlst_script.py`, `backup.properties`, `restore.properties`
			WLST commands:
			`exportWebCenterApplication`
			`importWebCenterApplication`

Prerequisites	WebCenter Portal must be installed, fully configured, and running on the target.	WebCenter Portal must be installed, fully configured, and running on the target.	WebCenter Portal must be installed, fully configured, and running on the target.
When to use	Use to back up and restore individual portals, portal hierarchies, and portal templates. Useful if only one or two portals or portal templates are corrupt.	Use to restore WebCenter Portal from a nightly/weekly backup that was previously taken using a backup script (in case of corruption). Use to restore configuration in `adf-config.xml`, `connections.xml`, and credentials in `/metadata/security/data/credentials`. Use to completely restore an entire WebCenter Portal installation on a new machine or WebLogic Server instance that is already installed and configured for Oracle WebCenter Portal.	Useful in a stage-to-production setup, where the production instance is installed and configured, and you want to copy WebCenter Portal on the stage instance (containing multiple portals, shared assets, security, and so on) to the target for the *first time*. Suitable for multi-site portals that use a large number of shared assets or other global artifacts that must be moved to the target in a single step. Not recommended for restoring a corrupt WebCenter Portal instance.
What is backed up / migrated	Portal-level customizations and user customizations for the portal. Portal-level task flow customizations, portal-level task flow instance customizations, and user customizations for portal level task flow instances. Portal content and portal data. Portal security permissions and roles. For details, see: Section 40.1.2.1, "Understanding Portal Archives" Section 40.1.3, "Deploying Portal Hierarchies" Section 40.2, "Deploying Portal Templates"	MDS metadata for all tools and services, such as discussions, announcements, events, portlets, activities, tags, worklists, and so on. Customizations of task flows, portlets, system pages, shared assets. Security roles and permissions for all portals and for global artifacts, as well as user-role assignments. Users and audit data are also migrated. Data stored in the WEBCENTER and MDS database schemas. Optionally, data stored in other schemas such as DISCUSSIONS, DISCUSSIONS_CRAWLER, ACTIVITIES, PORTLET, OCS, and so on.	MDS metadata for all tools and services, such as discussions, announcements, events, portlets, activities, tags, worklists, and so on. Customizations of task flows, portlets, system pages, shared assets. Security roles and permissions for all portals and for global artifacts, as well as user-role assignments Data stored in the WEBCENTER database schema for activity streams, portal events, feedback, lists, links, message boards, people connections, profiles, polls, surveys, and tags.

Table 41–1 (Cont.) Backup, Restore, and Migration Tools for WebCenter Portal

	Backup and Restore	Backup and Restore Scripts	Migration / Backup
	Portals / Portal Templates	**Full WebCenter Portal Install**	**WebCenter Portal Only**
What is not backed up / migrated	Application-level information, such as, shared assets, global customizations, and the Home portal.	WebCenter Portal domain.	Data stored on other back-end systems, such as the content server, discussions server, BPEL server, mail servers, and so on. WebCenter Portal connections to WSRP producers, PDK-Java producers, discussions servers, and so on. For details, see Section 40.1.2.1.3, "Understanding Connection Property Files." Application-level settings stored in `adf-config.xml` (domain/MDS) Credentials (metadata/security/data/credentials). WebCenter Portal domain.
Pros	Relatively quick as only specific portals or portal templates are backed up and restored. Allows more granular control over what is backed up and restored. Most efficient when only a few portals are corrupt.	Simple, extensible, and reliable way to regularly back up data owned by WebCenter Portal. Multiple, granular backup archives generated rather than a single large archive containing everything.	MDS data, WEBCENTER database data, customizations, and security captured in a single step. Simple to use and quicker than using four separate commands.
Cons	Cannot back up shared assets or the Home portal.	Database schemas WEBCENTER and MDS must be restored together. If not, data may become out-of-sync. If restoring additional schemas, such as OCS, you must restore them at the same time and from the same point to maintain data integrity. Incremental backup/restore is not supported. Domain configuration is not included in the backup script so you must back up the domain separately. See the "Backup and Recovery Recommendations for Oracle WebLogic Server" section in *Oracle Fusion Middleware Administrator's Guide*. Not recommended if you want to restore on a different instance with different back-end servers configured.	Requires a lot of internal processing. Native tools are not used to extract data from the database.

Note: Use Fusion Middleware test-to-production scripts to replicate a complete Fusion Middleware instance, installed and configured with WebCenter Portal, WebCenter Content, SOA Suite, BI, and so on, to one or more target environments. These scripts avoid you repeating complex install processes on multiple targets. For details, see the "Moving from a Test to a Production Environment" chapter in *Oracle Fusion Middleware Administrator's Guide*.

Test-to-production scripts are not recommended if the source WebCenter Portal installation has been used, that is, the customer has created metadata/data/security.

41.3 Backing Up Individual Portals

The back up process for one or more portals (or a portal hierarchy) is simple. You archive the portals and their content folders using the WLST command `exportWebCenterPortals` and then, if required, you back up any additional data that is stored for the portal in back-end components such as the discussions server.

The steps are as follows:

1. **Backup the portal to an export archive (PAR file)**.

   See Section 41.3.1, "Backing Up Portals Using WLST."

2. **Back up discussion data for the portal, if required**.

   See Section 41.3.2, "Backing Up Discussion Data for a Portal."

3. **Back up other external data, if any.**

   See Section 41.3.3, "Backing Up Other External Portal Data and Content."

The information in this section describes how to perform portal backups manually. If you need to back up frequently or want to set up a regular backup schedule, you can create a script that automates the back up process. For details, see Section 41.8, "Using Scripts to Back Up and Restore WebCenter Portal."

See also, Section 41.4, "Restoring Portals from a Backup."

> **Note:** The simultaneous back up of large numbers of portals is not recommended as, depending on server configuration, it may affect system performance. If a serious deterioration in performance is observed, break-down the backup/export process into several smaller groups.

41.3.1 Backing Up Portals Using WLST

Use the WLST command `exportWebCenterPortals` to back up a one or more portals to an archive (PAR file).

To find out what information is backed up inside a portal archive (PAR file) and what is not included, see Section 40.1.2.1, "Understanding Portal Archives."

> **Note:** Portal archives do not include shared assets or any information relating to the Home portal.

To prevent data loss, Oracle recommends that you:

- Take portals offline during the back up process to prevent data conflict (`offlineDuringExport=1`)
- Include portal content folders in the archive (`exportPortalContent=1`)
- Include connection information in the archive (`exportConnections=1`)

> **Note:** Connection information is not portal specific. All connections configured for the source WebCenter Portal installation are exported. See also, Section 40.1.2.1.3, "Understanding Connection Property Files."

- If a portal contains web service data controls or portlets, ensure that all associated web services or producers are up and accessible for the export to succeed.

For example, run the WLST command:

```
exportWebCenterPortals(appName='webcenter',
fileName='BackupSalesPortals_31March2013.par',
names='GlobalSales,MySales', offlineDuringExport=1,
exportPortalContent=1, exportConnections=1)
```

The options that you set depends on your specific archive requirements. For command syntax, see the "exportWebCenterPortals" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

To restore the portal at a later date, see Section 41.4, "Restoring Portals from a Backup."

41.3.2 Backing Up Discussion Data for a Portal

Use the Discussions Server Admin Console to back up discussion data for a specific portal to a `.zip` file that you restore later on, if required. For details, see Section 40.8.1.1, "Exporting Portal Discussions to an Archive" and Section 40.8.1.2, "Importing Portal Discussions from an Archive."

See Section 41.4, "Restoring Portals from a Backup."

41.3.3 Backing Up Other External Portal Data and Content

Backup files do not include externally stored data that portals reference through Content Presenter and Site Studio (such as external web content and pages) so you must back up external data separately. Similarly, if your portal references documents and files outside of its own content folder, you must ensure that all storage areas used by the portal are backed up.

In both cases, refer to the appropriate product documentation for instructions on how to back up the external data and content.

41.4 Restoring Portals from a Backup

You can restore one or more portals from a backup archive using the WLST command `importWebCenterPortals`. Existing portals are deleted and replaced.

The steps are as follows:

1. **Restore the portal, by importing the portal backup archive (PAR file) on the target**.

   See Section 41.4.1, "Restoring Portals from an Archive Using WLST."

2. **Restore discussion data for the portal, if required**.

   See Section 41.4.2, "Restoring Discussions Data for Portal."

3. **Restore other external data and content, if any.**

   See Section 41.4.3, "Restoring Other External Portal Data and Content."

The information in this section describes how to restore portal backups manually. If you prefer, you can create a script that automates the restoration process. For details, see Section 41.8, "Using Scripts to Back Up and Restore WebCenter Portal."

See also, Section 41.3, "Backing Up Individual Portals."

41.4.1 Restoring Portals from an Archive Using WLST

Use the WLST command `importWebCenterPortals` to restore one or more portals from an archive (PAR file).

To prevent data loss, Oracle recommends that you:

- Import connections used by the portal that are missing on the target, for some reason, before you restore the portal.

  See, Section 40.6.2, "Importing New WebCenter Portal Connections from a File."

- Take portals offline during portal restoration (`forceOffline=1`)

  Portal moderators can bring the portal back online after restoration.

- Import all the information inside the archive (`importCustomizations=1, importPortalContent=1, importSecurity=1, importData=1, importActivities=1`).

- If a portal contains web service data controls or portlets, all associated web services and producers must also be up and accessible for the import to succeed.

For example, run the WLST command:

```
importWebCenterPortals(appName='webcenter',
 fileName='BackupSalesPortals_31March2013.par', names='GlobalSales,MySales',
 parentPortal='Sales', importCustomizations=1, importPortalContent=1,
 importSecurity=1, importData=1, importActivities=1,
 overwrite=1, savePortals=1, forceOffline=1,
 importLog=/mybackups/RestoreSalesPortals_31march2013.log')
```

The options that you set depends on your specific requirements. For command syntax, see the "importWebCenterPortals" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Note: Portal-related data associated with some back-end components, specifically the discussions server, must be migrated after you export or import portals. See next topics, Section 41.4.2, "Restoring Discussions Data for Portal" and Section 41.4.3, "Restoring Other External Portal Data and Content."

41.4.2 Restoring Discussions Data for Portal

Use the Discussions Server Admin Console to restore discussion data for a particular portal from a backup `.zip` file. For details, see Section 40.8.1.2, "Importing Portal Discussions from an Archive" and Section 40.8.1.1, "Exporting Portal Discussions to an Archive."

See Section 41.3, "Backing Up Individual Portals."

41.4.3 Restoring Other External Portal Data and Content

If you backed up any external data or content that your portal uses, refer to the appropriate product documentation for instructions on how to restore information from your back ups, if required.

For example, you may want to regularly back up some externally stored data referenced by a portal through Content Presenter and Site Studio (such as external web content and pages) or documents that are stored outside the portal's own content folder.

41.5 Migrating Entire WebCenter Portal to Another Target

Using export and import, system administrators can migrate a WebCenter Portal instance to another target. This is useful in a stage-to-production setup, where the production instance is installed and configured and the entire WebCenter Portal instance on stage (containing multiple portals, shared assets, global artifacts, security, and so on) must be copied to the target for the first time.

You can also use the export and import utilities described in this section to back up global WebCenter Portal artifacts that are not owned by a particular portal, such as shared assets, business role pages, personal pages, and customized system pages.

This section includes the following topics:

- Section 41.5.1, "Understanding Import and Export for WebCenter Portal"
- Section 41.5.2, "Prerequisites for WebCenter Portal Export and Import"
- Section 41.5.3, "Exporting WebCenter Portal to an Archive"
- Section 41.5.4, "Importing a WebCenter Portal Archive"

41.5.1 Understanding Import and Export for WebCenter Portal

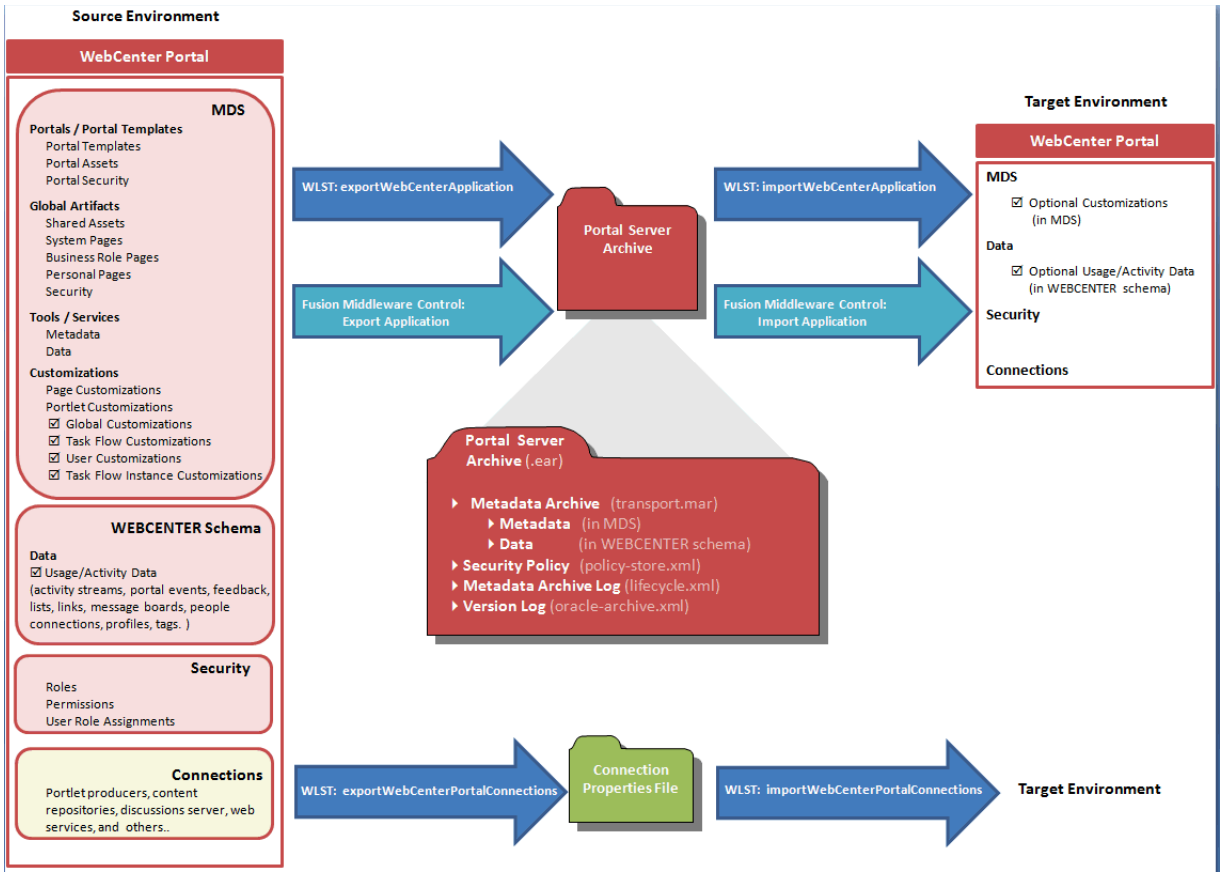
Using export and import, system administrators can migrate an entire WebCenter Portal instance between stage and production environments. You can export WebCenter Portal to a single export archive (`.ear` file) using WLST commands or Fusion Middleware Control, as shown in Figure 41–1.

The WebCenter Portal export archive (`.ear` file) contains several files, as listed in Table 41–1.

Table 41–2 Files in WebCenter Portal Export Archive EAR Files

Files	Description
`transport.mar`	Metadata archive that captures MDS metadata and database schema data for an entire WebCenter Portal instance
`policy-store.xml`	Security policy information.
`lifecycle.xml`	Log file that records details about the export process, such as which MDS paths are exported. The information in this file is useful if you want to compare two export archives or diagnose issues with the export and import process.
`oracle-archive.xml`	XML file containing version and compatibility information.

Figure 41–1 Migrating WebCenter Portal to Another Target



Information Included in a WebCenter Portal Archive

WebCenter Portal archives can include the following information that is stored in the metadata service (MDS) repository:

- **Portals and templates** - All portals and portal templates
- **Assets** - All shared assets and portal assets
- **Pages** - All pages, including system pages, business role pages, personal pages, and portal pages
- **Global customizations** - All global customizations applied to the application, system pages, other pages, assets, portlets, and task flows, that are stored in MDS.
- **User customizations** - All user-defined customizations applied to the application, pages, and task flows, that are stored in MDS. Sometimes referred to as *personalizations*.
- **Application and service metadata** - All application and service metadata stored in MDS (object definitions)

> **Note:** Some MDS customizations are *optional* on export, as illustrated in Figure 41–1. If you want to migrate all customizations you must set the export option "Include Customizations".

In addition, the WebCenter Portal archive (`.ear` file) can contain:

- **Tool/service data** - Database data associated with those tools and services that store data in the WebCenter Portal schema (WEBCENTER)

  Data migration is optional. To migrate data you must set the export option "Include Services Data".

- **Security** - All roles, permissions, and user role assignments:
  - application roles (and permissions assigned to each role)
  - users details and their application role assignments in the Home portal
  - individual portal members (and their role assignments in each portal)

Information Not Included in WebCenter Portal Archives

The WebCenter Portal archive (`.ear` file) does not include data associated with tools and services that do not store data in MDS or the WebCenter Portal database schema, such as analytics, activity graph, announcements, discussions, documents (on content server), instant messaging and presence (IMP), mail, pagelets, calendar events, personalizations, and worklists. To learn how to backup or move data associated with these tools and services, see Section 41.6, "Backing Up an Entire WebCenter Portal Installation."

Connection information is not included within the WebCenter Portal archive but you can export connection information configured in the source environment to a separate file and then deploy the connection information on the target. If some connection information, such as server names, ports, content management connections, and so on, varies between the two environments, you can isolate and modify the connection details before deploying the connection file. For details, see Section 40.6, "Moving Connections Details from Staging to Production."

Figure 41–2 lists in more detail exactly which information is always included, never included, or optional when you migrate WebCenter Portal to another target.

Figure 41–2 Information Exported and Imported with an Entire WebCenter Portal Instance

Always Exported and Imported	Optional on Import	Never Exported or Imported

MDS – Tool/Service Metadata
• Announcements
• Discussions
• Documents
• Events
• Lists (definitions)
• Notes
• Mail
• Pages
• Portlets
• Recent Activities
• Resource Catalog
• RSS News Feeds
• Search
• Tags
• Worklists
 	
MDS – Tool/Service Data
• Notes
 	
MDS - Shared / Portal Assets
• Page Templates
• Navigations
• Resource Catalogs
• Skins
• Page Styles
• Content Presenter Templates
• Mashup Styles
• Data Controls
• Task Flows
 	
Security Policy
• policy-store.xml:
  • Application roles and permissions
  • Portal roles and permissions
• User role assignments
	MDS – Global Customizations
	• Global administration Settings*
	• Shared asset customizations
	• System page customizations
	• Task flow customizations
	
	MDS – Portal Customizations
	• Portal administration settings
	• Portal asset customizations
	• Portal system page customizations
	• Task flow customizations
	
	MDS – Global Task Flow Customizations
	• All task flows
	
	MDS – Task Flow Instance Customizations
	• All task flows
	
	MDS – User Customizations
	• Pages
	• Task flows
	• Preferences
	
	WebCenter Portal Schema – Data
	• Activity Streams
	• Portal Events
	• Feedback
	• Links
	• Lists
	• Message Boards
	• People Connections
	• Profiles
	• Tags
	• Polls
	• People Connections:
	  • Default settings for profiles, message boards, feedback, connections, activity streams
	  • Activity stream task flow customizations
		External – Tool/Service Data
		• Documents (on Content Server)
		• Wikis and Blogs
		• Activity Graph
		• Analytics
		• Announcements
		• Discussions
		• IMP
		• Mail
		• Personal Events
		• Worklists
		
		External - Application Artifacts
		• Icons
		• Images...
		
		Out-of-the-box
		• Portal templates
		• Connections

* Except for people connection settings

WebCenter Portal export and import can be performed using Fusion Middleware Control or WLST commands. For details, see:

- Section 41.5.2, "Prerequisites for WebCenter Portal Export and Import"
- Section 41.5.3, "Exporting WebCenter Portal to an Archive"
- Section 41.5.4, "Importing a WebCenter Portal Archive"

41.5.2 Prerequisites for WebCenter Portal Export and Import

Before you export or import a WebCenter Portal instance, complete the following prerequisite tasks:

1. Back up or migrate all the back-end components *before* you export or import WebCenter Portal.

   Migrate back-end components for the application, such as the LDAP identity store, credential store, policy store, discussions server, content server, portlet producers, and so on. For more information, see Section 41.6, "Backing Up an Entire WebCenter Portal Installation."

2. Ensure that the database in which WebCenter Portal metadata and schema is stored is up and running otherwise export and import will not work.

3. If your application contains web service data controls or portlets, ensure that all associated web services or producers are up and accessible for export and import to succeed.

4. If you are migrating WebCenter Portal to another target, ensure that the tools and services configured in the target instance are a superset of the tools and services configured in the source instance. That is, the target must be configured with at least the same set of tools and services that the source is configured with. If this is not the case, the import operation fails.

5. Import connections exported from the source on to the target.

   For more information, see Section 40.6, "Moving Connections Details from Staging to Production."

6. Ensure that the users in both the source and target environment are identical.

   Important: If a shared identity store is not used, users must be migrated.

   Personal pages, that is, pages users create in the Home portal, are only migrated if the target and source applications both use the same LDAP identity store; this is because personal pages assignments are per user GUID.

   Verify that all users assigned the `Administrator` role in the source, exist in the target identity store. On import, users listed in WebCenter Portal's security policy are checked against the identity store that is configured for the domain. If a user is not found, any policies associated with that user are removed. See also, Section 31.4, "Moving the Administrator Account to an External LDAP Server."

7. Back up the `WEBCENTER` and `MDS` database schemas on the target before importing a WebCenter Portal archive.

   See Section 41.6, "Backing Up an Entire WebCenter Portal Installation."

8. Verify that the WebCenter Portal archive `.ear` file that you want to import was exported from WebCenter Portal 11.1.1.8.0 or later.

   You cannot import archives from earlier versions directly into WebCenter Portal 11.1.1.8.0 or later. If necessary, you must upgrade your source environment to 11.1.1.8.x before you create the export archive. For details, "Patching Oracle WebCenter Portal" in *Oracle Fusion Middleware Patching Guide*.

41.5.3 Exporting WebCenter Portal to an Archive

This section describes how to export an entire WebCenter Portal instance using Fusion Middleware Control and WLST commands. WebCenter Portal is exported into a single export archive (`.ear` file) that you can save to your local file system or to a remote server file system.

Note: For information about what the archive contains, see Section 41.1, "Understanding WebCenter Portal Back Up and Recovery."

This section includes the following:

- Section 41.5.3.1, "Exporting WebCenter Portal Using Fusion Middleware Control"
- Section 41.5.3.2, "Exporting WebCenter Portal Using WLST"

41.5.3.1 Exporting WebCenter Portal Using Fusion Middleware Control

System administrators can export an entire WebCenter Portal application using Fusion Middleware Control.

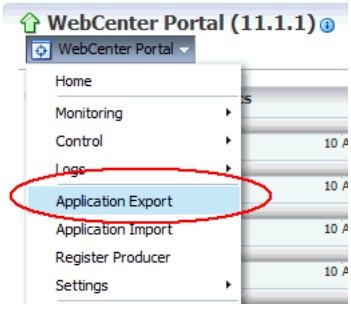
To export WebCenter Portal:

1. In Fusion Middleware Control, navigate to the home page for WebCenter Portal.

   See Section 6.2, "Navigating to the Home Page for WebCenter Portal."

2. From the **WebCenter Portal** menu, select **Application Export** (Figure 41–3).

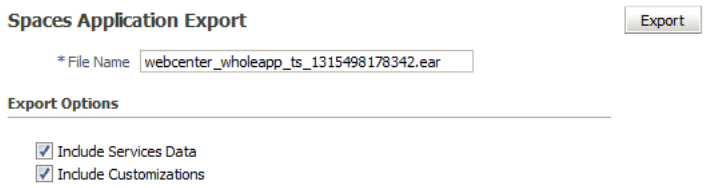
Figure 41–3 WebCenter Portal Menu - Application Export Option



3. Change the **File Name** for the export archive or accept the default name.

   To ensure uniqueness, the default `.ear` filename contains a unique ID—`webcenter_wholeapp_ts_`*`unique_ID`*`.ear` (Figure 41–4).

Figure 41–4 Naming the Export Archive



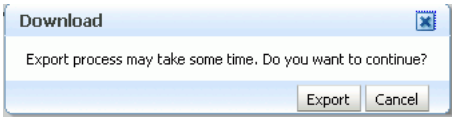
4. Set export options as required. For details, see Table 41–3.

Table 41–3 WebCenter Portal: Application Export Options

Field	Description
Include Tool/Service Data	Select to export data stored in the WebCenter Portal database schema (`WEBCENTER`) for the following tools and services: activity streams, portal events, feedback, lists, links, message boards, people connections, profiles, polls, surveys, and tags. Notes data stored in the MDS repository is exported too. Always re-export list data if source and target list definitions do not match. Mis-match only occurs when a list definition exists on the target and it is subsequently changed in the source. If the application you are exporting contains a large amount of data, consider using the database export utilities to export (and import) the WebCenter Portal schema data instead. For details see, Section 41.6.1.2, "Back Up (Export) WebCenter Portal Schema Data" and Section 41.6.1.3, "Restore (Import) WebCenter Portal Data." Deselect this option if you do not want to export data stored in the WebCenter Portal database schema (`WEBCENTER`). For example, if you are migrating WebCenter Portal from a test environment to a stage or production environment and the test data is no longer required. **Note:** The export process does *not* export data associated with services that store data externally such as analytics, activity graph, announcements, discussions, documents (on content server), instant messaging and presence (IMP), mail, pagelets, calendar events, personalizations, and worklists. To learn how to back up or move data associated with these services, see documentation for that product. See also, Section 41.6, "Backing Up an Entire WebCenter Portal Installation."
Include Customizations	Select to export application-level (global) customizations and user-level customizations. For information about which customizations are optional on export, see Figure 41–2. If you deselect this option, WebCenter Portal is exported without any application-level or user-level customizations. See also, Figure 41–2, "Information Exported and Imported with an Entire WebCenter Portal Instance".

5. Click **Export**.
6. In the Download dialog (Figure 41–5), click **Export** to confirm that you want to go ahead.

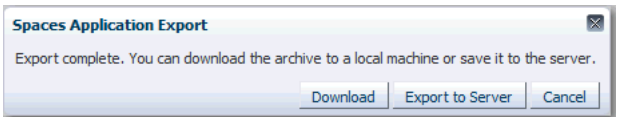
Figure 41–5 Downloading an Export Archive



Progress information is displayed during the export process. The application being exported cannot be accessed during export operations.

7. When the export process is complete, specify a location for the export archive (`.ear`).

Figure 41–6 Saving an Export Archive



Select one of:

- **Download** - Saves the export EAR file to your local file system.

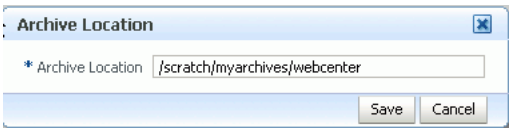
  Your browser downloads and saves the archive locally. The actual download location depends on your browser set up.

- **Export to Server** - Saves the export EAR file to a server location.

  When the Archive Location dialog displays (Figure 41–7), enter a suitable path for **Server Location**, for example, `/tmp`, and then click **Save**. The name of the EAR is not required here.

  Ensure that the server directory you specify has `write` permissions.

Figure 41–7 Archive Location



8. Click **Close** to dismiss the Export window.

The export archive (.EAR) is saved to the specified location.

Check the diagnostic log file, `WC_Spaces-diagnostic.log`, for any warnings or errors reported during the export process. To view the log file, select the menu option **WebCenter Portal > Logs > View Log Messages**. For details, see Section 28.2.1, "Viewing and Configuring WebCenter Portal Logs."

See also, Section G.7, "Troubleshooting WebCenter Portal Import and Export."

41.5.3.2 Exporting WebCenter Portal Using WLST

Use the WLST command `exportWebCenterApplication` to export an entire WebCenter Portal instance.

The following example exports WebCenter Portal together with all customizations in MDS (both application-level and user-level customizations) and database data to a file named `myAppExport.ear`.

```
wls:/weblogic/serverConfig>exportWebCenterApplication(appName='webcenter',
fileName='myAppExport.ear', exportCustomizations=1, exportData=1)
```

The following example exports a test WebCenter Portal instance. In this case, data created during testing (lists, events, links, tags, and so on) is not required in the target instance. The `.ear` file is saved to the location from which you run the WLST command:

```
wls:/weblogic/serverConfig>exportWebCenterApplication(appName='webcenter',
fileName='myTestAppExport.ear', exportData=0)
```

For command syntax and examples, see the "exportWebCenterApplication" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

41.5.4 Importing a WebCenter Portal Archive

This section describes how to import an entire WebCenter Portal application using Fusion Middleware Control and WLST commands.

Before importing WebCenter Portal, ensure that you complete all the tasks listed in Section 41.5.2, "Prerequisites for WebCenter Portal Export and Import."

This section includes the following:

- Section 41.5.4.1, "Importing WebCenter Portal Using Fusion Middleware Control"
- Section 41.5.4.2, "Importing WebCenter Portal Using WLST"
- Section 41.5.4.3, "Verifying WebCenter Portal After Import"

41.5.4.1 Importing WebCenter Portal Using Fusion Middleware Control

System administrators can import an entire WebCenter Portal instance using Fusion Middleware Control.

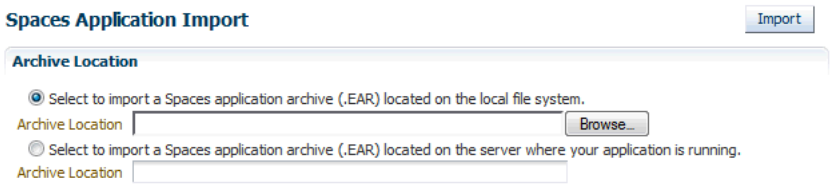
To import WebCenter Portal using Fusion Middleware Control:

1. In Fusion Middleware Control, navigate to the home page for WebCenter Portal.

   See Section 6.2, "Navigating to the Home Page for WebCenter Portal."

2. From the **WebCenter Portal** menu, select **Application Import**.
3. In the Application Import page (Figure 41–8), specify the location of your WebCenter Portal archive (`.ear`).

Figure 41–8 Application Import Page



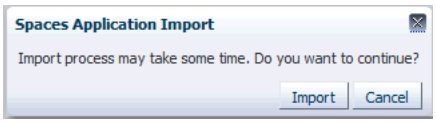
   Select one of the following:

   - **Archive Located on Local File System** - Enter the **Archive Location**. Alternatively, click **Browse** to locate the directory on the local file system where the `.ear` file is stored.
   - **Archive Located on Server File System** - Enter the **Archive Location**. Any shared location accessible from WebCenter Portal.

   The archive you select must contain an entire WebCenter Portal export—you cannot import individual portals or portal templates from here. Refer to Section 40.1.2.4, "Importing One or More Portals from an Archive" Section 40.1.2.4, "Importing One or More Portals from an Archive" for more information.

4. Click **Import**.
5. In the Application Import dialog (Figure 41–9), click **Import**.

Figure 41–9 Application Import dialog



Once the import is complete, a success message displays.

After importing an entire WebCenter Portal instance, log in to WebCenter Portal and verify the imported content. For details, see Section 41.5.4.3, "Verifying WebCenter Portal After Import."

41.5.4.2 Importing WebCenter Portal Using WLST

Use the WLST command `importWebCenterApplication` to import an entire WebCenter Portal instance from an archive. For command syntax and examples, see the "importWebCenterApplication" section in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

The following example imports WebCenter Portal from the export archive `myAppExport.ear`:

```
wls:/weblogic/serverConfig>importWebCenterApplication(appName='webcenter',fileName
='myAppExport.ear')
```

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

> **Note:** After importing the WebCenter Portal instance, log in to WebCenter Portal and verify the imported content. For details, see Section 41.5.4.3, "Verifying WebCenter Portal After Import."

41.5.4.3 Verifying WebCenter Portal After Import

After importing WebCenter Portal from an archive you must:

1. Restart the managed server (WC_Spaces) on which the newly imported WebCenter Portal instance is deployed.

   In a cluster environment, restart each managed server in the cluster. See also, Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

2. Log in to WebCenter Portal and verify that all portals and portal templates are available as expected.

   If not, see Section G.7.3, "Portals and Portal Templates Not Available After Import."

3. Initiate the Oracle Secure Enterprise Search crawler to index newly imported data.

   See also, *Oracle Secure Enterprise Search Administrator's Guide*.

41.6 Backing Up an Entire WebCenter Portal Installation

It is important to back up your entire WebCenter Portal installation on a frequent basis to avoid data loss due to database hardware failure or inadvertent removal of data from file or database.

This section outlines the steps required to completely back up all portals in the portal server, all database data, MDS, as well as data stored on other back-end servers. The back up process generates multiple, backup archives rather than a single large archive containing everything which facilitates a granular restore process.

The steps are as follows:

1. **Back up all data in the WebCenter Portal schema**.

   See Back Up (Export) WebCenter Portal Schema Data.

2. **Back up all data in the MDS schema.**

   See Back Up (Export) All MDS Schema Data.

3. **Back up all data for Content Server.**

   See Backing Up and Restoring All WebCenter Content Data.

4. **Back up all discussions server data.**

   See Backup (Export) All Discussions Schema Data.

5. **Back up other schema data stored for WebCenter Portal**.

   See Backing up and Restoring Other Schema Data (ACTIVITIES and PORTLET).

6. **Back up data for portlet producers used by WebCenter Portal.**

   See Backing Up and Restoring Portlet Producer Metadata.

7. **Back up pagelet producer metadata.**

   See Backing Up and Restoring Pagelet Producer Metadata.

8. **Back up activity graph and analytics metadata.**

   See Backing Up and Restoring Activity Graph and Analytics Metadata.

9. **Back up personalization data.**

   See Backing Up and Restoring Personalization Metadata.

10. **Back up security stores.**

    See Backing Up and Restoring LDAP Identity Store, Backing Up and Restoring Policy Stores (LDAP and Database) and Backing Up and Restoring Credential Stores (LDAP and Database).

11. **Back up the WebLogic domain hosting WebCenter Portal**.

    See Backing Up and Restoring a WebCenter Portal Domain.

12. **Back up Audit configuration.**

    See Backing Up and Restoring Audit Repository Configuration.

The information in this section describes how to back up manually. If you need to back up frequently or want to set up a regular backup schedule, you can create a script that automates the back up process. For details, see Section 41.8, "Using Scripts to Back Up and Restore WebCenter Portal."

41.6.1 Backing up and Restoring All WebCenter Portal Schema Data

WebCenter Portal's database schema (`WEBCENTER`) stores data for various tools and services including activity streams, portal events, feedback, lists, links, message boards, people connections, profiles, polls, surveys, and tags.

This section includes the following topics:

- Prerequisites
- Back Up (Export) WebCenter Portal Schema Data
- Restore (Import) WebCenter Portal Data

41.6.1.1 Prerequisites

If you are backing up or restoring an Oracle database schema, use `setenv` or `export` to set the following environment variables before backing up or restoring schema data:

- `ORACLE_HOME` - Database home
- `ORACLE_SID` - Service ID for the schemas
- `TNS_ADMIN` - Set to *`ORACLE_HOME`*`/network/admin`

41.6.1.2 Back Up (Export) WebCenter Portal Schema Data

To back up `WEBCENTER` schema data, use the appropriate utility for your database:

- For an Oracle database, go to *`DB_ORACLE_HOME`*`/bin` of your database and run the command described in Example 41–1, "Exporting WEBCENTER Schema Data".

  For detailed `expdp` command information, see *Oracle Database Utilities* guide.
- For non-Oracle databases, refer to the manufacturer's documentation.

Example 41–1 Exporting WEBCENTER Schema Data

```
sqlplus "sys/password as sysdba"
create or replace directory mydmpdirectory as
'full_path_to_directory_on_file_system';
GRANT read,write ON directory mydmpdirectory TO public;
exit;

DB_ORACLE_HOME/bin/expdp \"sys/password@serviceid as sysdba\"
 directory=mydmpdirectory dumpfile=webcenterportal.dmp SCHEMAS=srcprefix_WEBCENTER
 EXCLUDE=STATISTICS NOLOGFILE=Y
```

Where:

- `DB_ORACLE_HOME` is the directory in which the database for WebCenter Portal's schema (`WEBCENTER`) is installed.
- `password` is the password for the system database user.
- `serviceid` is the unique SID for the database. For example, `mydb1234`.
- `directory` is the location on the database machine where the dump file will be created.
- `dumpfile` is the name of the file that will contain the exported data.
- `SCHEMAS` identifies the target schema to be imported. Schema names include the RCU suffix that was used during installation (`_WEBCENTER`), along with a user supplied prefix. For example, `DEV_WEBCENTER`.
- `EXCLUDE=STATISTICS` specifies not to export statistics for the tables.
- `NOLOGFILE=Y` Suppresses the creation of a log file.

See also, Section 41.6.1.3, "Restore (Import) WebCenter Portal Data."

41.6.1.3 Restore (Import) WebCenter Portal Data

To restore WEBCENTER schema data from a backup, use the appropriate utility for your database:

- For an Oracle database, go to *DB_ORACLE_HOME*/bin of your database and run the command described in Example 41–2 or Example 41–3.

  For detailed impdp command information, see *Oracle Database Utilities* guide.

- For non-Oracle databases, refer to the manufacturer's documentation.

To restore the WEBCENTER schema on an Oracle database:

1. Shut down the target WebCenter Portal instance.
2. Go to *DB_ORACLE_HOME*/bin of the database where the WEBCENTER schema is installed, connect to the database using sqlplus as sysdba and run the following commands:

   ```
   DB_ORACLE_HOME/bin/sqlplus "sys/password@serviceid as sysdba"
   create or replace directory dmpdir as 'mydmpdirectory';
   GRANT read,write ON directory dmpdir TO public;
   ```

3. Do one of the following:

   - If schema names on the source and target match:

     ```
     drop user tgtprefix_WEBCENTER cascade;
     exit;
     ```

   - If schema names on the source and target are different:

     ```
     drop user tgtprefix_WEBCENTER cascade;
     create user tgtprefix_WEBCENTER identified by password default tablespace
     tgtprefix_IAS_WEBCENTER temporary tablespace name_IAS_TEMP;
     grant connect,resource to tgtprefix_WEBCENTER;
     exit;
     ```

   Where:

   - tgtprefix_WEBCENTER is the user name. This is the RCU suffix that was used during installation, _WEBCENTER, along with a user supplied prefix. For example, DEV_WEBCENTER.
   - password is the password for the target user.
   - tgtprefix_IAS_WEBCENTER identifies the default tablespace. For example, the RCU suffix that was used during installation, IAS_WEBCENTER, along with a user supplied prefix. For example, DEV_IAS_WEBCENTER.
   - name_IAS_TEMP identifies the temporary tablespace. For example, DEV_IAS_TEMP.

4. Run the import tool as described in Example 41–2 or Example 41–3.

Example 41–2 Importing WebCenter Portal Schema Data (Source and Target Schema Names Match)

```
DB_ORACLE_HOME/bin/impdp \"sys/password@serviceid as sysdba\"
 directory=mydmpdirectory dumpfile=webcenterportal.dmp SCHEMAS=tgtprefix_WEBCENTER
```

Example 41–3 Importing WebCenter Portal Schema Data (Source and Target Schema Names Different)

```
DB_ORACLE_HOME/bin/impdp \"sys/password@serviceid as sysdba\"
```

```
directory=mydmpdirectory dumpfile=webcenterportal.dmp
remap_schema=srcprefix_WEBCENTER:tgtprefix_WEBCENTER
remap_tablespace=source_tablespace:target_tablespace exclude=user
TABLE_EXISTS_ACTION=REPLACE
```

Where:

- `DB_ORACLE_HOME` is the directory in which the database for WebCenter Portal's schema (`WEBCENTER`) is installed.
- `password` is the password for the system database user.
- `serviceid` is the unique SID for the database. For example, `mydb1234`.
- `directory` is the location on the database machine where the dump file is located.
- `dumpfile` is the name of the file that contains data to be imported.
- `SCHEMAS` identifies the target schema to be imported. Schema names include the RCU suffix that was used during installation (`_WEBCENTER`), along with a user supplied prefix. For example, `DEV_WEBCENTER`.

  Use this parameter when schema names on the source and target match. For example, both schemas are named `DEV_WEBCENTER`.
- `REMAP_SCHEMA` identifies the source and target schemas. Use this parameter when schema names on the source and target are different. Schema names include the RCU suffix that was used during installation, `_WEBCENTER`, along with the user supplied prefix. For example, `DEV_WEBCENTER`.
- `REMAP_TABLESPACE` identifies the source and target tablespace. Remaps all objects selected for import with persistent data in the source tablespace to be created in the target tablespace. For example, source_tablespace:target_tablespace.
- `TABLE_EXISTS_ACTION=REPLACE` drops the current table and creates the table as it is in the dump file.

41.6.2 Backing Up and Restoring All MDS Schema Data

The `MDS` schema contains customization metadata and data for WebCenter Portal.

This section includes the following topics:

- Prerequisites
- Back Up (Export) All MDS Schema Data
- Restore (Import) MDS Schema Data

41.6.2.1 Prerequisites

If you are backing up or restoring an Oracle database schema, use `setenv` or `export` to set the following environment variables before backing up or restoring schema data:

- `ORACLE_HOME` - Database home
- `ORACLE_SID` - Service ID for the schemas
- `TNS_ADMIN` - Set to *`ORACLE_HOME`*`/network/admin`

Note: For these back up (export) and restore (import) procedures to work, the schema names on the source and target *must* match. For example, both schemas must be named `DEV_MDS`.

41.6.2.2 Back Up (Export) All MDS Schema Data

To back up `MDS` data, use the appropriate utility for your database:

- For an Oracle database, go to *`DB_ORACLE_HOME`*`/bin` of your database and run the command described in Example 41–4, "Exporting MDS Schema Data".

  For detailed `expdp` command information, see *Oracle Database Utilities* guide.
- For non-Oracle databases, refer to the manufacturer's documentation.

Example 41–4 Exporting MDS Schema Data

```
sqlplus "sys/password as sysdba"
create or replace directory mydmpdirectory as
'full_path_to_directory_on_file_system';
GRANT read,write ON directory mydmpdirectory TO public;
exit;

DB_ORACLE_HOME/bin/expdp \"sys/password@serviceid as sysdba\"
 directory=mydmpdirectory dumpfile=mds.dmp SCHEMAS=srcprefix_MDS
 EXCLUDE=STATISTICS NOLOGFILE=Y
```

Where:

- `DB_ORACLE_HOME` is the directory in which the database for WebCenter Portal's MDS schema is installed.
- `password` is the password for the system database user.
- `serviceid` is the unique SID for the database. For example, `mydb1234`.
- `directory` is the location on the database machine where the dump file will be created.
- `dumpfile` is the name of the file that will contain the exported data.
- `SCHEMAS` is the schema to be exported. Include the RCU suffix that was used during installation (`_MDS`), along with a user supplied prefix. For example, `DEV_MDS`.

  Schema names on the source and target *must* match. For example, both schemas must be named `DEV_MDS`.
- `EXCLUDE=STATISTICS` specifies not to export statistics for the tables.
- `NOLOGFILE=Y` Suppresses the creation of a log file.

See also, Section 41.6.2.3, "Restore (Import) MDS Schema Data."

41.6.2.3 Restore (Import) MDS Schema Data

To restore `MDS` schema data from a backup, use the appropriate utility for your database:

- For an Oracle database, go to *`DB_ORACLE_HOME`*`/bin` of your database and run the command described in Example 41–5, "Importing MDS Schema Data".

  For detailed `impdp` command information, see *Oracle Database Utilities* guide.

- For non-Oracle databases, refer to the manufacturer's documentation.

To restore the MDS schema on an Oracle database:

1. Shut down the target MDS instance.
2. Go to *DB_ORACLE_HOME*/bin of the database where the MDS schema is installed, connect to the database using sqlplus as sysdba and run the following commands:

```
DB_ORACLE_HOME/bin/sqlplus "sys/password@serviceid as sysdba"
create or replace directory dmpdir as 'mydmpdirectory';
GRANT read,write ON directory dmpdir TO public;
```

3. Drop the MDS schema and exit sqlplus:

```
drop user tgtprefix_MDS cascade;
exit;
```

4. Run the import tool as described in Example 41–5, "Importing MDS Schema Data".

Example 41–5 Importing MDS Schema Data

```
DB_ORACLE_HOME/bin/impdp \"sys/password@serviceid as sysdba\"
 directory=mydmpdirectory dumpfile=mds.dmp SCHEMA=tgtprefix_MDS
```

Where:

- DB_ORACLE_HOME is the directory in which the database for WebCenter Portal's MDS schema is installed.
- password is the password for the system database user.
- serviceid is the unique SID for the database. For example, mydb1234.
- directory is the location on the database machine where the dump file is located.
- dumpfile is the name of the file that contains data to be imported.
- SCHEMAS is the schema to be imported. Include the RCU suffix that was used during installation (_MDS), along with the user supplied prefix. For example, DEV_MDS.

  Schema names on the source and target *must* match. For example, both schemas must be named DEV_MDS.

41.6.3 Backing Up and Restoring All WebCenter Content Data

To fully back up Oracle WebCenter Content, you must back up data the WebCenter Content database schema (OCS), back up all the WebCenter Content native (vault) and web-viewable (weblayout) files, and also back up other configuration data. For details, see the "Backup and Recovery Recommendations for Oracle WebCenter Content" section in *Oracle Fusion Middleware Administrator's Guide*.

Optionally, you can back up the root folder for a WebCenter Portal instance (or specific Portal Framework application) to a separate archive. A root folder backup may be useful if the folder becomes corrupt or you want to migrate the entire the folder to another target. For detailed instructions, see the "System Migration and Archiving" section in *Oracle Fusion Middleware Administering Oracle WebCenter Content*.

Note: Consider the following when restoring or migrating root folders for WebCenter Portal:

- **Security data is not archived with the root folder**
- **Root folder migration must take place before you start WebCenter Portal for the first time**

  (WebCenter Portal only). When you start WebCenter Portal for the first time a root folder is automatically created for WebCenter Portal on the Content Server. You cannot later overwrite this folder with a root folder archive exported from a *different* WebCenter Portal instance as internal root folder IDs will not match. If you plan to migrate root folder content, you must do so *before* the WebCenter Portal instance starts up for the first time.

- **Folder ID "counter" on source and target must match**

  Every time you create a folder on Content Server, a folder ID counter increments by one. If the counter on the source and target is not in sync you may experience issues when you try to create folders on the target after an import operation. For example, if the folder ID counter on the target is on 4 when you import folders with IDs 5,6,7,8, you will see an error the next time you try to create a folder on the target as it will attempt to create a folder with an ID of 5. The only workaround is to manually alter the counter table on the target using SQL.

As root folder backups are not appropriate for every restoration use case, Oracle recommends full WebCenter Content database schema back ups for your primary back up/restore strategy.

After restoring WebCenter Content data, log in to WebCenter Portal and open any portal that utilizes document-related task flows. Verify that document services are enabled in that portal and that imported folders are available as expected.

41.6.4 Backing up and Restoring Discussion Schema Data

Discussions and announcements store information in two database schemas:

- `DISCUSSIONS`: stores discussions and announcements data
- `DISCUSSIONS_CRAWLER`: enables Oracle Secure Enterprise Search (SES) to crawl the discussions server

This section includes the following topics:

- Prerequisites
- Backup (Export) All Discussions Schema Data
- Restore (Import) Discussions Schema Data

41.6.4.1 Prerequisites

If you are backing up or restoring an Oracle database schema, use `setenv` or `export` to set the following environment variables before backing up or restoring schema data:

- `ORACLE_HOME` - Database home
- `ORACLE_SID` - Service ID for the database

- TNS_ADMIN - Set to *ORACLE_HOME*/network/admin

41.6.4.2 Backup (Export) All Discussions Schema Data

To back up all discussions schema data, use the appropriate utility for your database:

- For an Oracle database, go to *DB_ORACLE_HOME*/bin of your database and run the command described in Example 41–6, "Exporting Discussions Schema Data".

  For detailed expdp command information, see *Oracle Database Utilities* guide.

- For non-Oracle databases, refer to the manufacturer's documentation.

Notes:

- This section describes how to export all discussions server data. If you want to export discussions for a single portal, see Section 41.3.2, "Backing Up Discussion Data for a Portal."

Example 41–6 Exporting Discussions Schema Data

```
sqlplus "sys/password as sysdba"
create or replace directory mydmpdirectory as
'full_path_to_directory_on_file_system';
GRANT read,write ON directory mydmpdirectory TO public;
exit;

DB_ORACLE_HOME/bin/expdp \"sys/password@serviceid as sysdba\"
directory=mydmpdirectory dumpfile=discussions.dmp
SCHEMAS=srcprefix_DISCUSSIONS,srcprefix_DISCUSSIONS_CRAWLER EXCLUDE=STATISTICS
NOLOGFILE=Y
```

Where:

- DB_ORACLE_HOME is the directory in which the database for WebCenter Portal's discussions schemas are installed.
- password is the password for the system database user.
- serviceid is the unique SID for the database. For example, mydb1234.
- directory is the location on the database machine where the dump file will be created.
- dumpfile is the name of the file that will contain the exported data.
- SCHEMAS identifies the schemas to be exported. Include the RCU suffix that was used during installation (_DISCUSSIONS and _DISCUSSIONS_CRAWLER), along with a user supplied prefix. For example, DEV_DISCUSSIONS.

  To export data from both schemas, separate each schema name with a comma.

- EXCLUDE=STATISTICS specifies not to export statistics for the tables.
- NOLOGFILE=Y Suppresses the creation of a log file.

See also, Section 41.6.4.3, "Restore (Import) Discussions Schema Data."

41.6.4.3 Restore (Import) Discussions Schema Data

To restore discussions schema data from a backup, use the appropriate utility for your database:

- For an Oracle database, go to *DB_ORACLE_HOME*/bin of your database and run the command described in Example 41–7 or Example 41–8.

  For detailed impdp command information, see *Oracle Database Utilities* guide.

- For non-Oracle databases, refer to the manufacturer's documentation.

To restore DISCUSSIONS and DISCUSSIONS_CRAWLER schemas on an Oracle database:

1. Shut down the target discussions server.
2. Go to *DB_ORACLE_HOME*/bin of the database where the DISCUSSIONS and DISCUSSIONS_CRAWLER schema is installed, connect to the database using sqlplus as sysdba and run the following commands:

   ```
   DB_ORACLE_HOME/bin/sqlplus "sys/password@serviceid as sysdba"
   create or replace directory dmpdir as 'mydmpdirectory';
   GRANT read,write ON directory dmpdir TO public;
   ```

3. Do one of the following:

   - If schema names on the source and target match:

     ```
     drop user tgtprefix_DISCUSSIONS cascade;
     drop user tgtprefix_DISCUSSIONS_CRAWLER cascade;
     exit;
     ```

   - If schema names on the source and target are different:

     ```
     drop user tgtprefix_DISCUSSIONS cascade;
     drop user tgtprefix_DISCUSSIONS_CRAWLER cascade;
     create user tgtprefix_DISCUSSIONS identified by password default tablespace
     tgtprefix_IAS_DISCUSSIONS temporary tablespace name_IAS_TEMP;
     grant connect,resource to tgtprefix_DISCUSSIONS
     exit;
     ```

   Where:

   - tgtprefix_DISCUSSIONS is the user name. This is the RCU suffix that was used during installation, _DISCUSSIONS, along with a user supplied prefix. For example, DEV_DISCUSSIONS.
   - password is the password for the target user.
   - tgtprefix_IAS_DISCUSSIONS identifies the default tablespace. For example, the RCU suffix that was used during installation, IAS_DISCUSSIONS, along with a user supplied prefix. For example, DEV_IAS_DISCUSSIONS.
   - name_IAS_TEMP identifies the temporary tablespace. For example, DEV_IAS_TEMP.

4. Run the import tool as described in Example 41–7 or Example 41–8.

Example 41–7 Importing Discussions Schema Data (Source and Target Schema Names Match)

```
DB_ORACLE_HOME/bin/impdp \"sys/password@serviceid as sysdba\"
 directory=mydmpdirectory dumpfile=discussions.dmp
 SCHEMAS=tgtprefix_DISCUSSIONS,tgtprefix_DISCUSSIONS_CRAWLER
```

Example 41–8 Importing Discussions Schema Data (Source and Target Schema Names Different)

```
DB_ORACLE_HOME/bin/impdp \"sys/password@serviceid as sysdba\"
 directory=mydmpdirectory dumpfile=discussions.dmp
 remap_schema=srcprefix_DISCUSSIONS:tgtprefix_DISCUSSIONS
 remap_schema=srcprefix_DISCUSSIONS_CRAWLER:tgtprefix_DISCUSSIONS_CRAWLER
 remap_tablespace=source_tablespace:target_tablespace exclude=user
 TABLE_EXISTS_ACTION=REPLACE
```

Where:

- DB_ORACLE_HOME is the directory in which the database for WebCenter Portal's discussions schemas are installed.
- password is the password for the system database user.
- serviceid is the unique SID for the database. For example, mydb1234.
- directory is the location on the database machine where the dump file is located.
- dumpfile is the name of the file that contains data to be imported.
- SCHEMAS identifies the schema (or schemas) to be imported. Include the RCU suffix that was used during installation (_DISCUSSIONS and _DISCUSSIONS_CRAWLER), along with a user supplied prefix. The DISCUSSIONS and DISCUSSIONS_CRAWLER schemas have the same user supplied prefix, for example, DEV_DISCUSSIONS and DEV_DISCUSSIONS_CRAWLER.

  Use this parameter when schema names on the source and target match. For example, schemas in the source and target database are both named DEV_DISCUSSIONS and DEV_DISCUSSIONS_CRAWLER.
- REMAP_SCHEMA identifies the source and target schemas. Use this parameter when schema names on the source and target are different. Schema names include the RCU suffix that was used during installation, _DISCUSSIONS, along with the user supplied prefix. For example, DEV_DISCUSSIONS.

  The DISCUSSIONS and DISCUSSIONS_CRAWLER schemas have the same user supplied prefix.
- REMAP_TABLESPACE identifies the source and target tablespace. Remaps all objects selected for import with persistent data in the source tablespace to be created in the target tablespace. For example, source_tablespace:target_tablespace.
- TABLE_EXISTS_ACTION=REPLACE drops the current table and creates the table as it is in the dump file.

41.6.5 Backing up and Restoring Other Schema Data (ACTIVITIES and PORTLET)

In addition to the schemas mentioned in the previous topic (WEBCENTER, MDS, DISCUSSIONS, and DISCUSSIONS_CRAWLER), WebCenter Portal can store data is several other schemas:

- ACTIVITIES Stores activity graph and analytics data
- PORTLET Stores portlet and pagelet data

The backup and restore procedures are common for all schemas. Use the appropriate utility for your database:

- For non-Oracle databases, refer to the manufacturer's documentation.
- For an Oracle database, go to *DB_ORACLE_HOME*/bin of your database and run the commands described in:
  - Example 41–9, "Exporting Schema Data (Oracle Database)"
  - Example 41–10, "Importing Schema Data (Source and Target Schema Names Match)"
  - Example 41–11, "Importing Schema Data (Source and Target Schema Names Different)"

  For detailed expdp and impdp command information, see *Oracle Database Utilities* guide.

Prerequisites (Oracle Database)

If you are backing up or restoring an Oracle database schema, use setenv or export to set the following environment variables before backing up or restoring schema data:

- ORACLE_HOME - Database home
- ORACLE_SID - Service ID for the schemas
- TNS_ADMIN - Set to *ORACLE_HOME*/network/admin

Exporting Schema Data (Oracle Database)

Example 41–9 shows a sample expdp command for exporting Oracle database schema data. Replace schemadump.dmp and SCHEMA_NAME to match the schema you want to export.

Example 41–9 Exporting Schema Data (Oracle Database)

```
sqlplus "sys/password as sysdba"
create or replace directory mydmpdirectory as
'full_path_to_directory_on_file_system';
GRANT read,write ON directory mydmpdirectory TO public;
exit;

DB_ORACLE_HOME/bin/expdp \"sys/password@serviceid as sysdba\"
 directory=mydmpdirectory dumpfile=schemadump.dmp SCHEMAS=srcprefix_SCHEMA_NAME
 EXCLUDE=STATISTICS NOLOGFILE=Y
```

Where:

- DB_ORACLE_HOME is the directory in which the database schema is installed.
- password is the password for the system database user.
- serviceid is the unique SID for the database. For example, mydb1234.
- directory is the location on the database machine where the dump file will be created.
- dumpfile is the name of the file that will contain the exported data.
- SCHEMAS is the schema (or schemas) to be exported. This is the RCU suffix that was used during installation (*_SCHEMA_NAME*), along with the user supplied prefix. For example, DEV_ACTIVITIES.

  If you want to export data from multiple schemas, separate each schema name with a comma.
- EXCLUDE=STATISTICS specifies not to export statistics for the tables.

- NOLOGFILE=Y suppresses the creation of a log file.

Importing Schema Data (Oracle Database)

Example 41–10 and Example 41–11 show sample impdp commands for importing schema data. Replace schemadump.dmp and SCHEMA_NAME to match the schema you want to import.

1. Shut down the target WebCenter Portal instance.
2. Go to *DB_ORACLE_HOME*/bin of the database where the schema is installed, connect to the database using sqlplus as sysdba and run the following commands:

   ```
   DB_ORACLE_HOME/bin/sqlplus "sys/password@serviceid as sysdba"
   create or replace directory dmpdir as 'mydmpdirectory';
   GRANT read,write ON directory dmpdir TO public;
   ```

3. Do one of the following:
   - If schema names on the source and target match:

     ```
     drop user tgtprefix_SCHEMA_NAME cascade;
     exit;
     ```

   - If schema names on the source and target are different:

     ```
     drop user tgtprefix_SCHEMA_NAME cascade;
     create user tgtprefix_SCHEMA_NAME identified by password default tablespace
     tgtprefix_IAS_SCHEMA_NAME temporary tablespace name_IAS_TEMP;
     grant connect,resource to tgtprefix_SCHEMA_NAME;
     exit;
     ```

   Where:

   - tgtprefix_SCHEMA_NAME is the user name. This is the RCU suffix that was used during installation, _SCHEMA_NAME, along with a user supplied prefix. For example, DEV_ACTIVITIES.
   - password is the password for the target user.
   - tgtprefix_IAS_SCHEMA_NAME identifies the default tablespace. For example, the RCU suffix that was used during installation, IAS_SCHEMA_NAME, along with a user supplied prefix. For example, DEV_IAS_ACTIVITIES.
   - name_IAS_TEMP identifies the temporary tablespace. For example, DEV_IAS_TEMP.
4. Run the import tool as described in Example 41–10 or Example 41–11.

Example 41–10 Importing Schema Data (Source and Target Schema Names Match)

```
DB_ORACLE_HOME/bin/impdp \"sys/password@serviceid as sysdba\"
 directory=mydmpdirectory dumpfile=schemadump.dmp SCHEMAS=tgtprefix_SCHEMA_NAME
```

Example 41–11 Importing Schema Data (Source and Target Schema Names Different)

```
DB_ORACLE_HOME/bin/impdp \"sys/password@serviceid as sysdba\"
 directory=mydmpdirectory dumpfile=schemadump.dmp
 remap_schema=srcprefix_SCHEMA_NAME:tgtprefix_SCHEMA_NAME
 remap_tablespace=source_tablespace:target_tablespace exclude=user
 TABLE_EXISTS_ACTION=REPLACE
```

Where:

- `DB_ORACLE_HOME` is the directory in which the database schema is installed.
- `password` is the password for the system database user.
- `serviceid` is the unique SID for the database. For example, `mydb1234`.
- `directory` is the location on the database machine where the dump file is located.
- `dumpfile` is the name of the file that contains data to be imported.
- `SCHEMAS` is the schema (or schemas) to be imported. This is the RCU suffix that was used during installation (*_SCHEMA_NAME*), along with the user supplied prefix. For example, `DEV_ACTIVITIES`.

  Use this parameter when schema names on the source and target match. For example, both schemas must be named `DEV_ACTIVITIES`.

  If you want to export data from multiple schemas, separate each schema name with a comma.
- `REMAP_SCHEMA` identifies the source and target schemas. Use this parameter when schema names on the source and target are different. Schema names include the RCU suffix that was used during installation, `_SCHEMA_NAME`, along with the user supplied prefix. For example, `DEV_ACTIVITIES`.
- `REMAP_TABLESPACE` identifies the source and target tablespace. Remaps all objects selected for import with persistent data in the source tablespace to be created in the target tablespace. For example, `source_tablespace:target_tablespace`.
- `TABLE_EXISTS_ACTION=REPLACE` drops the current table and creates the table as it is in the dump file.

41.6.6 Backing Up and Restoring LDAP Identity Store

External identity stores, such as Oracle Internet Directory, store data in the underlying database. For information on how to back up and restore database schema data for Oracle Internet Directory, see the "Backup and Recovery Recommendations for Oracle Internet Directory" section in *Oracle Fusion Middleware Administrator's Guide*.

If you are using a different LDAP identity store, refer to the appropriate back up and recovery documentation for that product.

41.6.7 Backing Up and Restoring Policy Stores (LDAP and Database)

Use the WLST command `migrateSecurityStore` to back up and then restore the policy store that is configured for WebCenter Portal or your Portal Framework application. In a production environment, Oracle recommends that policies are stored in LDAP or a database. File-based policy stores are *not* recommended.

Use `migrateSecurityStore` to:

- Back up your LDAP or database-based policy store to a backup file
- Restore your LDAP or database policy store from a backup file

For details, see the "Migrating Policies Manually" section in *Oracle Fusion Middleware Application Security Guide*.

See also, the "migrateSecurityStore" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

Note: Security policy data is included when you use WebCenter Portal's export/import utilities (`exportWebCenterApplication` and `importWebCenterApplication`) to migrate WebCenter Portal to another instance so there is no need to manually migrate the policy store in this instance. For more information, see Section 41.5, "Migrating Entire WebCenter Portal to Another Target."

41.6.8 Backing Up and Restoring Credential Stores (LDAP and Database)

Use the WLST command `migrateSecurityStore` to back up and then restore the credential store that is configured for WebCenter Portal or your Portal Framework application. In a production environment, Oracle recommends that credentials are stored in LDAP or a database. File-based credential stores are *not* recommended.

Use `migrateSecurityStore` to:

- Back up your LDAP or database-based credential store to a backup file
- Restore your LDAP or database credential store from a backup file

For details, see the "Migrating Credentials with migrateSecurityStore" section in *Oracle Fusion Middleware Application Security Guide*.

See also, "migrateSecurityStore" in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

41.6.9 Backing Up and Restoring a WebCenter Portal Domain

For information on how to back up and restore your domain configuration, see the "Backup and Recovery Recommendations for Oracle WebLogic Server" section in *Oracle Fusion Middleware Administrator's Guide*.

41.6.10 Backing Up and Restoring Portlet Producer Metadata

Portlet producers can store registration handles and portlet preference data as metadata with the consumer application, that is, WebCenter Portal or a Portal Framework application. This section describes how to back up any portlet metadata that is stored by your application using the WLST command `exportPortletClientMetadata` and how to restore the portlet metadata using `importPortletClientMetadata`.

Note: Portlet metadata is included when you use WebCenter Portal's export/import utilities (`exportWebCenterApplication` and `importWebCenterApplication`) to migrate WebCenter Portal to another instance so there is no need to manually migrate portlet producer metadata in this instance. For more information, see Section 41.5, "Migrating Entire WebCenter Portal to Another Target."

This section includes the following topics:

- Backing Up (Exporting) Portlet Client Metadata
- Restoring (Importing) Portlet Client Metadata

For information on how to back up portlet producer data stored on the database, see o Section 41.6.5, "Backing up and Restoring Other Schema Data (ACTIVITIES and PORTLET)."

41.6.10.1 Backing Up (Exporting) Portlet Client Metadata

To export portlet client metadata and producer customizations and personalizations, for a single application, such as WebCenter Portal or a Portal Framework application, use the WLST command `exportPortletClientMetadata`. This command exports metadata for all the portlet producers used by the application. You cannot opt to export metadata for specific producers.

For detailed syntax and examples, see the "exportPortletClientMetadata" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

41.6.10.2 Restoring (Importing) Portlet Client Metadata

To import portlet client metadata and producer customizations and personalizations, for WebCenter Portal or a Portal Framework application, use the WLST command `importPortletClientMetadata`.

Prerequisites:

- The database in which the application metadata or schema is stored and the portlet producers must be up and running.
- Use the WLST command `exportPortletClientMetadata` to export the portlet client metadata and producer customizations and personalizations to an `.ear` file, See also, Section 41.6.10.1, "Backing Up (Exporting) Portlet Client Metadata."

For detailed syntax and examples, see the "importPortletClientMetadata" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*. See also, the "Metadata Services (MDS) Custom WLST Commands" section.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

41.6.11 Backing Up and Restoring Pagelet Producer Metadata

The pagelet producer stores configuration data and content in MDS. You can back up pagelet metadata to a separate archive using the `exportMetadata` and `importMetadata` WLST commands. For details, see Section 22.5, "Managing Import, Export, Backup and Recovery of Pagelet Producer Components."

41.6.12 Backing Up and Restoring Activity Graph and Analytics Metadata

Activity graph stores metadata definitions for mapping WebCenter Portal event data from analytics in MDS. You can back up activity graph metadata to a separate archive using the `exportAGMetadata` and `importAGMetadata` WLST commands. You can also back up provider configuration metadata, for a given provider, to an activity graph metadata definition file using the WLST command `exportAGProviderConfiguration`. For details, see Section 10.6, "Managing Activity Graph Schema Customizations."

To back up the entire ACTIVITIES database schema, see Section 41.6.5, "Backing up and Restoring Other Schema Data (ACTIVITIES and PORTLET)."

41.6.13 Backing Up and Restoring Personalization Metadata

Personalization files are stored in MDS. You can back up personalization data stored in MDS for WebCenter Portal or your Portal Framework application to an archive using

the `exportMetadata` WLST command and restore personalization data from the archive, if required, using `importMetadata`.

To back up (export) personalization data in MDS:

```
exportMetadata(application='wcps-services', server='server_name',
 toLocation='archive_file_name', remote='true')
```

To restore (import) personalization data from an archive to MDS:

```
importMetadata(application='wcps-services', server='server_name',
 fromLocation='archive_file_name', remote='true')
```

For example:

```
exportMetadata(application='wcps-services', server='WC_Utilities',
 toLocation='/backup_11_11_2013/backupWCPS.zip', remote='true')

importMetadata(application='wcps-services', server='WC_Utilities',
 fromLocation='/backup_11_11_2013/backupWCPS.zip, remote='true')
```

41.6.14 Backing Up and Restoring Audit Repository Configuration

You can back up audit policies and audit repository configuration to a file using the `exportAuditConfig` and `importAuditConfig` WLST commands.

For detailed syntax and examples, see the "exportAuditConfig" and "importAuditConfig" sections in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

41.7 Restoring an Entire WebCenter Portal Installation

This section describes how to restore your WebCenter Portal installation after some hardware failure or inadvertent removal of data from file or database. Use the steps in this section to completely restore an entire WebCenter Portal installation on a new machine or WebLogic Server instance that is already installed and configured for Oracle WebCenter Portal.

The steps in this section assume that the back-end servers and connections used in the restored instance are exactly the same as those configured prior to the restoration process.

Important: Database schemas `WEBCENTER` and `MDS` *must* be restored together to ensure the data is in-sync.

If you need to restore additional schemas, such as OCS, you must restore them at the same time and from the same point to maintain data integrity.

The steps are as follows:

1. **Restore WebCenter Portal schema from a backup**.

   See Restore (Import) WebCenter Portal Data.

2. **Restore MDS schema data from a backup.**

   See Restore (Import) MDS Schema Data.

3. **(Optional) Restore Content Server data from a backup.**

See Backing Up and Restoring All WebCenter Content Data.

4. **(Optional) Restore discussion schema data from a backup.**

   See Restore (Import) Discussions Schema Data.

5. **(Optional) Restore other schemas data for WebCenter Portal from a backup**.

   See Backing up and Restoring Other Schema Data (ACTIVITIES and PORTLET).

6. **Restore security store data from backups.**

   For details, see:

   - Backing Up and Restoring Policy Stores (LDAP and Database)
   - Backing Up and Restoring Credential Stores (LDAP and Database)
   - (Optional) Backing Up and Restoring LDAP Identity Store

7. **(Optional) Restore connections for WebCenter Portal from a backup.**

   See Importing New WebCenter Portal Connections from a File.

8. **(Optional) Restore audit configuration for WebCenter Portal from a backup.**

   See Backing Up and Restoring Audit Repository Configuration.

9. **(Optional) Restore the WebLogic Server domain hosting WebCenter Portal from a backup**.

   See Backing Up and Restoring a WebCenter Portal Domain.

10. Restart, and verify restored

In some situations you may need to restore metadata associated with individual tools and services. In this case, refer to the following topic:

- **Restore only portlet producer metadata from a backup.**

  See Backing Up and Restoring Portlet Producer Metadata.

- **Restore only pagelet producer MDS metadata from a backup.**

  See Backing Up and Restoring Pagelet Producer Metadata.

- **Restore only activity graph and analytics MDS metadata from a backup.**

  See Backing Up and Restoring Activity Graph and Analytics Metadata.

- **Restore only personalization data in MDS from a backup.**

  See Backing Up and Restoring Personalization Metadata.

The information in this section describes how to restore manually. If you need to restore or migrate data frequently, you can create a script that automates the process. For details, see Section 41.8, "Using Scripts to Back Up and Restore WebCenter Portal."

41.8 Using Scripts to Back Up and Restore WebCenter Portal

Backing up your WebCenter Portal installation manually can take time. Using scripts to automate and schedule regular back ups is more efficient and saves a great deal of time. To help you get started, Oracle provides a sample backup script that you can customize to suit your installation and back up requirements.

For more information, read the following topics:

- Section 41.8.1, "Understanding Back Up and Restore Script Files"

- Section 41.8.2, "Using Scripts to Back Up WebCenter Portal"
- Section 41.8.3, "Restoring WebCenter Portal from Backups Using Scripts"

41.8.1 Understanding Back Up and Restore Script Files

Oracle provides sample scripts to help automate your back up and recovery processes. The sample scripts back up and restore the following information:

- **Database schemas:** Back up all the required schemas for WebCenter Portal.
- **Data in file stores:** Back up and restore WebCenter Portal data stored in the WebCenter Content file system.
- **Security information:** Back up and restore policy store, credential store, and audit configuration for WebCenter Portal.

Table 41–4 describes the sample scripts and files provided for back up and recovery:

Table 41–4 Sample Scripts and Files for Back up and Restore

Sample Scripts and Files	Description	Use to...
`master_script.sh`	Shell script that executes database export commands, archives WebCenter Content on the file system, and executes WLST export and import commands. See Section 41.8.1.1, "master_script.sh."	Back up and restore
`wlst_script.py`	Python script that runs WLST commands for exporting and importing portlet and security metadata. See Section 41.8.1.2, "wlst_script.py."	Back up and restore
`backup.properties`	Properties file that contains input parameters to back up WebCenter Portal databases and run WLST export commands in `master_script.sh` and `wlst_script.py`. See Section 41.8.1.3, "backup.properties and restore.properties Files."	Back up only
`restore.properties`	Properties file that contains input parameters for `master_script.sh` and `wlst_script.py` that enable you to restore WebCenter Portal databases and run WLST import commands from backup files. See Section 41.8.1.3, "backup.properties and restore.properties Files."	Restore only

The sample files are starter scripts for you to review and modify. Alternatively, you can create your own scripts from scratch, if preferred.

41.8.1.1 master_script.sh

`master_script.sh` is shown in Example 41–12.

This script can back up (export) WebCenter Portal data stored in the following database schemas:

- `WEBCENTER`
- `MDS`
- `DISCUSSIONS`

- DISCUSSIONS_CRAWLER
- OCS
- ACTIVITIES
- PORTLET

During back up, the script executes an export database command expdp for each schema you want to back up:

```
DB_ORACLE_HOME/bin/expdp \"sys/password@serviceid as sysdba\"
directory=backup_directory dumpfile=dump_file_name.dmp SCHEMAS=prefix_SCHEMA_NAME
EXCLUDE=STATISTICS NOLOGFILE=y
```

> **See Also:** The expdp database command for individual schemas are described in Section 41.6, "Backing Up an Entire WebCenter Portal Installation."

The script also exports or imports WebCenter Content native files (vault folder) and web-viewable files (weblayout folder) stored on the file system.

- To back up WebCenter Content files stored on the file system, the script executes the following:

  ```
  tar cvf wcc_vault.tar WCP_ORACLE_HOME/ucm/vault
  ```

  ```
  tar cvf wcc_weblayout.tar WCP_ORACLE_HOME/ucm/weblayout
  ```

- To restore WebCenter Content files on the target file system, the script executes the following:

  ```
  tar xvf wcc_vault.tar
  ```

  ```
  tar xvf wcc_weblayout.tar
  ```

Finally, the script calls the WLST command script wlst_script.py. For details, see Section 41.8.1.2, "wlst_script.py."

Example 41–12 Sample Script - master_script.sh

```
## master_script.sh
## Backs up or restores a WebCenter Portal installation
## Executes database export or import commands and a Python script containing WLST
commands.
############ No User Input Required ########################################
# Reading the properties files for WebCenter Portal back up or restore...
PROPS_FILE=$1

exportimport=`sed '/^\#/d' $PROPS_FILE | grep 'OPERATION'  | tail -n 1 | cut -d
"=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
dump_directory=`sed '/^\#/d' $PROPS_FILE | grep 'DATA_DIRECTORY'  | tail -n 1 |
cut -d "=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
oracle_db_home=`sed '/^\#/d' $PROPS_FILE | grep 'DB_ORACLE_HOME'  | tail -n 1 |
cut -d "=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
oracle_db_admin=`sed '/^\#/d' $PROPS_FILE | grep 'DB_ADMIN_USER'  | tail -n 1 |
cut -d "=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
oracle_db_adminpwd=`sed '/^\#/d' $PROPS_FILE | grep 'DB_ADMIN_PASSWORD'  | tail -n
1 | cut -d "=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
oracle_db_sid=`sed '/^\#/d' $PROPS_FILE | grep 'DB_SID'  | tail -n 1 | cut -d "="
-f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
oracle_db_connect_webcenter=`sed '/^\#/d' $PROPS_FILE | grep
```

```
'DB_CONNECT_WEBCENTER_SCHEMA'  | tail -n 1 | cut -d "=" -f2- | sed
's/^[[:space:]]*//;s/[[:space:]]*$//'`
oracle_db_connect_mds=`sed '/^\#/d' $PROPS_FILE | grep 'DB_CONNECT_MDS_SCHEMA'  |
tail -n 1 | cut -d "=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
oracle_db_connect_discussions=`sed '/^\#/d' $PROPS_FILE | grep
'DB_CONNECT_DISCUSSIONS_SCHEMA'  | tail -n 1 | cut -d "=" -f2- | sed
's/^[[:space:]]*//;s/[[:space:]]*$//'`
oracle_db_connect_ocs=`sed '/^\#/d' $PROPS_FILE | grep 'DB_CONNECT_OCS_SCHEMA'  |
tail -n 1 | cut -d "=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
oracle_db_connect_activities=`sed '/^\#/d' $PROPS_FILE | grep
'DB_CONNECT_ACTIVITIES_SCHEMA'  | tail -n 1 | cut -d "=" -f2- | sed
's/^[[:space:]]*//;s/[[:space:]]*$//'`
oracle_db_connect_portlet=`sed '/^\#/d' $PROPS_FILE | grep
'DB_CONNECT_PORTLET_SCHEMA'  | tail -n 1 | cut -d "=" -f2- | sed
's/^[[:space:]]*//;s/[[:space:]]*$//'`

#Read schema information from the properties file.
src_webcenter_schema=`sed '/^\#/d' $PROPS_FILE | grep 'EXP_WEBCENTER_SCHEMA'  |
tail -n 1 | cut -d "=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
src_mds_schema=`sed '/^\#/d' $PROPS_FILE | grep 'EXP_MDS_SCHEMA'  | tail -n 1 |
cut -d "=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
src_ocs_schema=`sed '/^\#/d' $PROPS_FILE | grep 'EXP_OCS_SCHEMA'  | tail -n 1 |
cut -d "=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
src_discussions_schema=`sed '/^\#/d' $PROPS_FILE | grep 'EXP_DISCUSSIONS_SCHEMA'
| tail -n 1 | cut -d "=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
src_discussions_crawler_schema=`sed '/^\#/d' $PROPS_FILE | grep
'EXP_DISCUSSIONS_CRAWLER_SCHEMA'  | tail -n 1 | cut -d "=" -f2- | sed
's/^[[:space:]]*//;s/[[:space:]]*$//'`
src_activities_schema=`sed '/^\#/d' $PROPS_FILE | grep 'EXP_ACTIVITIES_SCHEMA'  |
tail -n 1 | cut -d "=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
src_portlet_schema=`sed '/^\#/d' $PROPS_FILE | grep 'EXP_PORTLET_SCHEMA'  | tail
-n 1 | cut -d "=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`


# Read WLST connection information from the properties file.
username=`sed '/^\#/d' $PROPS_FILE | grep 'WLST_ADMIN_USER'  | tail -n 1 | cut -d
"=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
password=`sed '/^\#/d' $PROPS_FILE | grep 'WLST_ADMIN_PASSWORD'  | tail -n 1 | cut
-d "=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
adminconsole=`sed '/^\#/d' $PROPS_FILE | grep 'WLST_ADMIN_CONSOLE'  | tail -n 1 |
cut -d "=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
wlstlocation=`sed '/^\#/d' $PROPS_FILE | grep 'WLST_LOCATION'  | tail -n 1 | cut
-d "=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
wlstscriptfile=`sed '/^\#/d' $PROPS_FILE | grep 'WLST_SCRIPT_LOCATION'  | tail -n
1 | cut -d "=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
wcpServer=`sed '/^\#/d' $PROPS_FILE | grep 'WCP_SERVER_NAME'  | tail -n 1 | cut -d
"=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
jpsConfigFile=`sed '/^\#/d' $PROPS_FILE | grep 'JPS_CONFIG_FILE'  | tail -n 1 |
cut -d "=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
sourceJpsContextPolicy=`sed '/^\#/d' $PROPS_FILE | grep
'SRC_JPS_CONTEXT_POLICYSTORE'  | tail -n 1 | cut -d "=" -f2- | sed
's/^[[:space:]]*//;s/[[:space:]]*$//'`
destinationJpsContextPolicy=`sed '/^\#/d' $PROPS_FILE | grep
'TGT_JPS_CONTEXT_POLICYSTORE'  | tail -n 1 | cut -d "=" -f2- | sed
's/^[[:space:]]*//;s/[[:space:]]*$//'`
sourceJpsContextCred=`sed '/^\#/d' $PROPS_FILE | grep 'SRC_JPS_CONTEXT_CREDSTORE'
| tail -n 1 | cut -d "=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
destinationJpsContextCred=`sed '/^\#/d' $PROPS_FILE | grep
'TGT_JPS_CONTEXT_CREDSTORE'  | tail -n 1 | cut -d "=" -f2- | sed
's/^[[:space:]]*//;s/[[:space:]]*$//'`
```

```
backupPolicyStoreFile=`sed '/^\#/d' $PROPS_FILE | grep 'POLICYSTORE_FILE_NAME'  |
tail -n 1 | cut -d "=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
backupCredStoreFile=`sed '/^\#/d' $PROPS_FILE | grep 'CREDSTORE_FILE_NAME'  | tail
-n 1 | cut -d "=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
wccVaultLoc=`sed '/^\#/d' $PROPS_FILE | grep 'WCC_VAULT_LOC'  | tail -n 1 | cut -d
"=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`
wccWeblayoutLoc=`sed '/^\#/d' $PROPS_FILE | grep 'WCC_WEBLAYOUT_LOC'  | tail -n 1
| cut -d "=" -f2- | sed 's/^[[:space:]]*//;s/[[:space:]]*$//'`

#Data dump files that database schema data is exported to or imported from
wcdmp=wcdmp.dmp
mdsdmp=mdsdmp.dmp
discussionsdmp=discussionsdmp.dmp
ocsdmp=ocsdmp.dmp
activitiesdmp=activities.dmp
portletdmp=portlet.dmp

#Portlet client metadata export archive (.EAR) that portlet client metadata is
exported to or imported from
portletdatafilename=portletdata.ear

#Audit configuration file that audit information is exported to or imported from
auditFileName=audit.xml

#Running WebCenter Portal back up and recovery scripts...

#On backup - Create a folder with a timestamp under the dump_directory folder
#On restore - Read user specified base directory to import from
current_time=$(date "+%Y.%m.%d-%H.%M.%S")
backup_directory=$dump_directory
if [ ! -z "$exportimport" ]; then
  if [ $exportimport = 'export' ]; then
    #'Creating backup directory.'
    backup_directory=$dump_directory/$current_time
    rm -rf $backup_directory
    mkdir $backup_directory
  fi
  if [ $exportimport = 'import' ]; then
    backup_directory=$dump_directory
  fi
fi

#Writing output to a log file
outputLogFile=$2
# Create a pipe file
mknod $backup_directory/pipefile.$$ p
# Start tee process in background to read it and output content to screen and log
file
rm -rf $backup_directory/$outputLogFile
tee $backup_directory/$outputLogFile <$backup_directory/pipefile.$$ &
exec &>$backup_directory/pipefile.$$


#Common for backup (export) and restore (import)
#Create directories and grant read write permissions
export ORACLE_HOME=$oracle_db_home
export ORACLE_SID=$oracle_db_sid
export TNS_ADMIN=$ORACLE_HOME/network/admin
cd $oracle_db_home/bin
```

```
if [ ! -z "$exportimport" ]; then
  # Start back up (export)
  if [ $exportimport = 'export' ]; then
    echo 'Back up started...'
    if [ -n "$src_webcenter_schema" ] && [ -n "$wcdmp" ]; then
      ./sqlplus "$oracle_db_connect_webcenter as sysdba" << eof_disp
      create or replace directory dmpdir as '$backup_directory';
      GRANT read,write ON directory dmpdir TO public;
eof_disp
      echo 'Exporting the WEBCENTER  schema...'
      ./expdp \"$oracle_db_connect_webcenter as sysdba\" directory=dmpdir
 dumpfile=$wcdmp  SCHEMAS=$src_webcenter_schema EXCLUDE=STATISTICS NOLOGFILE=y
    fi
    if [ -n "$src_mds_schema" ] && [ -n "$mdsdmp" ]; then
      ./sqlplus "$oracle_db_connect_mds as sysdba" << eof_disp
      create or replace directory dmpdir as '$backup_directory';
      GRANT read,write ON directory dmpdir TO public;
eof_disp
      echo 'Exporting the MDS schema...'
      ./expdp \"$oracle_db_connect_mds as sysdba\" directory=dmpdir
 dumpfile=$mdsdmp  SCHEMAS=$src_mds_schema EXCLUDE=STATISTICS NOLOGFILE=y
    fi
    if [ -n "$src_discussions_schema" ] && [ -n "$discussionsdmp" ]; then
      ./sqlplus "$oracle_db_connect_discussions as sysdba" << eof_disp
      create or replace directory dmpdir as '$backup_directory';
      GRANT read,write ON directory dmpdir TO public;
eof_disp
      echo 'Exporting the DISCUSSIONS schema...'
      ./expdp \"$oracle_db_connect_discussions as sysdba\" directory=dmpdir
 dumpfile=$discussionsdmp  SCHEMAS=$src_discussions_schema EXCLUDE=STATISTICS
 NOLOGFILE=y
    fi
    if [ -n "$src_ocs_schema" ] && [ -n "$ocsdmp" ]; then
      ./sqlplus "$oracle_db_connect_ocs as sysdba" << eof_disp
      create or replace directory dmpdir as '$backup_directory';
      GRANT read,write ON directory dmpdir TO public;
eof_disp
      echo 'Exporting the OCS schema...'
      ./expdp \"$oracle_db_connect_ocs as sysdba\" directory=dmpdir
 dumpfile=$ocsdmp  SCHEMAS=$src_ocs_schema EXCLUDE=STATISTICS NOLOGFILE=y
     if [ -n "$wccVaultLoc" ]; then
       echo -e '\nExporting vault files for WebCenter Content...'
       cd $backup_directory
       tar cvf wcc_vault.tar -C $wccVaultLoc/vault .
       if [ -f "$backup_directory/wcc_vault.tar" ]; then
         echo -e '\nExported vault files for WebCenter Content to:
 '$backup_directory'/wcc_vault.tar'
       fi
       cd $oracle_db_home/bin
     fi
     if [ -n "$wccWeblayoutLoc" ]; then
       echo -e '\nExporting weblayout files for WebCenter Content...'
       cd $backup_directory
       tar cvf wcc_weblayout.tar -C $wccWeblayoutLoc/weblayout .
       if [ -f "$backup_directory/wcc_weblayout.tar" ]; then
         echo -e '\nExported weblayout files for WebCenter Content to:
 '$backup_directory'/wcc_weblayout.tar'
       fi
       cd $oracle_db_home/bin
     fi
```

```
    fi
    if [ -n "$src_activities_schema" ] && [ -n "$activitiesdmp" ]; then
      ./sqlplus "$oracle_db_connect_ocs as sysdba" << eof_disp
      create or replace directory dmpdir as '$backup_directory';
      GRANT read,write ON directory dmpdir TO public;
eof_disp
      echo 'Exporting the ACTIVITIES schema...'
      ./expdp \"$oracle_db_connect_activities as sysdba\" directory=dmpdir
 dumpfile=$activitiesdmp  SCHEMAS=$src_activities_schema EXCLUDE=STATISTICS
 NOLOGFILE=y
    fi
    if [ -n "$src_portlet_schema" ] && [ -n "$portletdmp" ]; then
      ./sqlplus "$oracle_db_connect_ocs as sysdba" << eof_disp
      create or replace directory dmpdir as '$backup_directory';
      GRANT read,write ON directory dmpdir TO public;
eof_disp
      echo 'Exporting the PORTLET schema...'
      ./expdp \"$oracle_db_connect_portlet as sysdba\" directory=dmpdir
 dumpfile=$portletdmp  SCHEMAS=$src_portlet_schema EXCLUDE=STATISTICS NOLOGFILE=y
    fi

    #Call the WLST command script.
    cd $wlstlocation
    ./wlst.sh $wlstscriptfile $exportimport $username $password $adminconsole
 $backup_directory/$portletdatafilename $wcpServer $jpsConfigFile
 $sourceJpsContextPolicy $destinationJpsContextPolicy $sourceJpsContextCred
 $destinationJpsContextCred $backup_directory/$auditFileName

#Copy the backup policy store and credential store files to the backup location.
    if [ -f "$backupPolicyStoreFile" ]; then
      mv $backupPolicyStoreFile $backup_directory
    fi
    if [ -f "$backupCredStoreFile" ]; then
      mv $backupCredStoreFile $backup_directory
    fi
    echo 'Back up completed successfully. Backup created at location:
 '$backup_directory'. Check the log file: '$backup_directory/$outputLogFile' for
additional details.'
  fi

  #Start restore (import)...
  if [ $exportimport = 'import' ]; then
    echo 'Restore started...'
    if [ -f "$backup_directory/wcc_vault.tar" ]; then
        echo -e '\nImporting vault files for WebCenter Content...'
        cd $wccVaultLoc/vault
        tar xvf $backup_directory/wcc_vault.tar
        echo -e '\nImported vault files for WebCenter Content from:
 '$backup_directory'/wcc_vault.tar to the location: '$wccVaultLoc'/vault'
    fi
      if [ -f "$backup_directory/wcc_weblayout.tar" ]; then
        echo -e '\nImporting weblayout files for WebCenter Content...'
        cd $wccWeblayoutLoc/weblayout
        tar xvf $backup_directory/wcc_weblayout.tar
        echo -e '\nImported weblayout files for WebCenter Content from:
 '$backup_directory'/wcc_weblayout.tar to the location:
 '$wccWeblayoutLoc'/weblayout'
      fi
    #Call the WLST commands script.
    cd $wlstlocation
```

```
    ./wlst.sh $wlstscriptfile $exportimport $username $password $adminconsole
 $backup_directory/$portletdatafilename $wcpServer $jpsConfigFile
 $destinationJpsContextPolicy $sourceJpsContextPolicy $destinationJpsContextCred
 $sourceJpsContextCred $backup_directory/$auditFileName
    echo 'Restoration completed successfully. Check the log file:
 '$backup_directory/$outputLogFile' for additional details.'
  fi
fi
#Clean up pipe file
rm -f $backup_directory/pipefile.$$
```

41.8.1.2 wlst_script.py

`wlst_script.sh` is shown in Example 41–13.

This script connects to the Admin Console for your WebCenter Portal installation, and then either backs up (exports) or restores (imports) the following:

- Portlet client metadata
- Policy store
- Credential store
- Audit configuration information

Export WLST Commands Executed During Back Up

During back up, the script executes the following WLST export commands:

- `exportPortletClientMetadata(appName, fileName, server)`
- `migrateSecurityStore(type='appPolicies', configFile, src, dst, overWrite, srcApp, dstApp)`
- `migrateSecurityStore(type='credStore', configFile, src, dst)`
- `exportAuditConfig(fileName)`

Import WLST Commands Executed During Restore

During restore, the script executes the following WLST import commands:

- `importPortletClientMetadata(appName, fileName, server)`
- `migrateSecurityStore(type='appPolicies', configFile, src, dst, overWrite, srcApp, dstApp)`
- `migrateSecurityStore(type='credStore', configFile, src, dst)`
- `importAuditConfig(fileName)`

> **See Also:** If you want to back up or restore individual items, refer to the appropriate section in Section 41.6, "Backing Up an Entire WebCenter Portal Installation" or Section 41.7, "Restoring an Entire WebCenter Portal Installation."

Example 41–13 Sample Script - wlst_script.py

```
## wlst_script.py
## Python script that runs export and import WLST commands.
############ No User Input Required ###################

# Get user credentials and other parameters from the properties file
```

```
exportOrImport = sys.argv[1]
username = sys.argv[2]
password = sys.argv[3]
adminconsole = sys.argv[4]
fileName = sys.argv[5]
wcpServerName = sys.argv[6]
jpsConfigFile = sys.argv[7]
destination = sys.argv[8]
source = sys.argv[9]
dstCred = sys.argv[10]
sourceCred = sys.argv[11]
auditFileName=sys.argv[12]

# Connect to the given host
connect(username,password,adminconsole)

if (exportOrImport == 'export' ):
# Run export WLST commands
  # Export portlet data
  print 'Exporting portlet data...'
  exportPortletClientMetadata(appName='webcenter', fileName=fileName,
server=wcpServerName)
  if webcenterErrorOccurred(): # COMMAND STATUS
      print "Error while exporting the portlet data."
  else:
      print 'Successfully exported the portlet data.'

  # Export security
  print 'Exporting the policy store...'
  migrateSecurityStore(type='appPolicies', configFile=jpsConfigFile, src=source,
 dst=destination, overWrite='true', srcApp='webcenter', dstApp='webcenter')
  print 'Exporting the credential store...'
  migrateSecurityStore(type='credStore', configFile=jpsConfigFile, src=sourceCred,
 dst=dstCred)
  print 'Exporting audit configuration...'
  exportAuditConfig(fileName=auditFileName)

elif (exportOrImport == 'import' ):
# Run import WLST commands
  # Import portlet data
  print 'Importing portlet data...'
  importPortletClientMetadata(appName='webcenter', fileName=fileName,
server=wcpServerName)
  if webcenterErrorOccurred(): # COMMAND STATUS
      print "Error while importing portlet data."
  else:
      print 'Successfully imported portlet data.'

  # Import security
  print 'Importing the policy store...'
  migrateSecurityStore(type='appPolicies', configFile=jpsConfigFile, src=source,
dst=destination, overWrite='true', srcApp='webcenter', dstApp='webcenter')
  print 'Importing the credential store...'
  migrateSecurityStore(type='credStore', configFile=jpsConfigFile, src=sourceCred,
dst=dstCred)
  print 'Importing audit configuration...'
  importAuditConfig(fileName=auditFileName)
```

41.8.1.3 backup.properties and restore.properties Files

The `backup.properties` file contains input parameters for backup commands in master_script.sh and wlst_script.py. For example, file names, database home location, database connect string, schema names, and so on.

A similar `.properties` file (`restore.properties`) is required to define input parameters for restore commands.

Table 41–5 lists and describes the input parameters in `backup.properties` and `restore.properties` files.

Example 41–14 shows a `backup.properties` file with sample values.

Example 41–15 shows a `restore.properties` file with sample values.

Table 41–5 User Defined Parameters for Back Up and Restore Scripts

Back up / Restore Parameter	Description	Example

OPERATION	Determines whether the script backs up WebCenter Portal data (exports) or restores WebCenter Portal data (imports).	**For back up:**
		`export`
		For restore:
		`import`

Database information		

DATA_DIRECTORY	**For back up scripts:**
	Location on the file system under which backup files created by the script are stored.
	Each time you run the script, a new subdirectory is created under the directory specified here. The name of each subdirectory includes a timestamp, such as `2013.03.18-05.20.28`.
	For restore scripts:
	Directory containing the back up you want to restore from.
		For back up:
		`DATA_DIRECTORY=/scratch/aime1/mywebcenterportal_backupscripts/mybackups`
		For restore:
		`DATA_DIRECTORY=/scratch/aime1/mywebcenterportal_backupscripts/mybackups/2013.03.18-05.20.28`

DB_ORACLE_HOME	Database home directory.	`/scratch/aime1/mywork/db1234`
DB_ADMIN_USER	Database admin user.	`mydbadminuser`
DB_ADMIN_PASSWORD	Password for the database admin user.	`mypassword`
DB_SID	Database SID.	`db1234`
WebCenter Content folders		**For back up and restore:**
WCC_VAULT_LOC	Location on the file system for WebCenter Content vault files.	`/scratch/aime1/mwork/mymw/user_projects/domains/WLS_WC/ucm/cs`
WCC_WEBLAYOUT_LOC	Location of the file system for WebCenter Content weblayout files.	/scratch/aime1/mwork/mymw/user_projects/domains/WLS_WC/ucm/cs
Database connect strings (Back up scripts only)	Required when OPERATION=export.	**For back up only:**
DB_CONNECT_WEBCENTER_SCHEMA	Connect string for the WEBCENTER database schema you want to export.	`mydbadmin/mypassword@db1234`
DB_CONNECT_MDS_SCHEMA	Connect string for the MDS database schema you want to export.	`mydbadmin/mypassword@db1234`
DB_CONNECT_OCS_SCHEMA	Connect string for the OCS database schema you want to export.	`mydbadmin/mypassword@db1234`
DB_CONNECT_DISCUSSIONS_SCHEMA	Connect string for the DISCUSSIONS database schema you want to export.	`mydbadmin/mypassword@db1234`
DB_CONNECT_ACTIVITIES_SCHEMA	Connect string for the ACTIVITIES database schema you want to export.	`mydbadmin/mypassword@db1234`

Table 41–5 (Cont.) User Defined Parameters for Back Up and Restore Scripts

Back up / Restore Parameter	Description	Example
DB_CONNECT_PORTLET_SCHEMA	Connect string for the PORTLET database schema you want to export.	`mydbadmin/mypassword@db1234`
Database schemas to export (Back up scripts only)	Required when OPERATION=export.	**For back up only:**
EXP_WEBCENTER_SCHEMA	Name of the WEBCENTER schema to export.	`mysrcprefix_WEBCENTER`
EXP_MDS_SCHEMA	Name of the MDS schema to export.	`mysrcprefix_MDS`
EXP_DISCUSSIONS_SCHEMA	Name of the DISCUSSIONS schema to export.	`mysrcprefix_DISCUSSIONS`
EXP_DISCUSSIONS_CRAWLER_SCHEMA	Name of the DISCSUSSIONS_CRAWLER schema to export.	`mysrcprefix_DISCUSSIONS_CRAWLER`
EXP_OCS_SCHEMA	Name of the OCS schema to export.	`mysrcprefix_OCS`
EXP_ACTIVITIES_SCHEMA	Name of the ACTIVITIES schema to export.	`mysrcprefix_ACTIVITIES`
EXP_PORTLET_SCHEMA	Name of the PORTLET schema to export.	`mysrcprefix_PORTLET`
WLST Export and Import		**For back up and restore:**
WLST - General		
WLST_ADMIN_USER	Name of the administrative user connecting WLST to the Administration Server.	`mywlstadmin`
WLST_ADMIN_PASSWORD	Password of the administrative user.	
WLST_ADMIN_CONSOLE	Host name and port of the Administration Server, specified using the format: *protocol*`://`*listen_address*`:`*listen_port*	`t3://myhost.com:24647`
WLST_LOCATION	Location of the WLST script. You must run all Oracle WebCenter Portal WLST commands from your WebCenter Portal Oracle home directory (WCP_ORACLE_HOME): `WCP_ORACLE_HOME/common/bin/wlst.sh`	`/scratch/aime1/mywork/mymw/mywcp_oraclehome/common/bin`
WLST_SCRIPT_LOCATION	Location of the WLST back up and restore script.	`/scratch/aime1/myportal_server_scripts/wlst_script.py`
WCP_SERVER_NAME	Name of the managed server on which the WebCenter Portal application (`webcenter`) is deployed.	`WC_Spaces`
WLST - Security		
JPS_CONFIG_FILE	Name and location of the configuration file (by default, named `jps-config.xml`) relative to the directory where the WLST command is run.	`/scratch/aime1/mywork/mymw/user_projects/domains/myDomainHome/config/fmwconfig/backup-config-mycopy.xml`
SRC_JPS_CONTEXT_POLICYSTORE	Name of a `jps-context` in the configuration file, where the source policy store is specified.	`mysourcePolicy`
TGT_JPS_CONTEXT_POLICYSTORE	Name of another `jps-context` in the configuration file, where the target policy store is specified.	`mytargetPolicy`
SRC_JPS_CONTEXT_CREDSTORE	Name of a `jps-context` in the configuration file, where the source credential store is specified.	`mysourceCred`
TGT_JPS_CONTEXT_CREDSTORE	Name of another `jps-context` in the configuration file, where the target credential store is specified.	`mytargetCred`
POLICYSTORE_FILE_NAME	Name and location of the policy store that you want to back up or restore (as specified in JPS_CONFIG_FILE)	`/scratch/portal_server_scripts/backup/backup-system-jazn-data.xml`
CREDSTORE_FILE_NAME	Name and location of the credential store that you want to back up ore restore (location is as specified in JPS_CONFIG_FILE, with the file name cwallet.sso)	`/scratch/portal_server_scripts/backup/cwallet.sso`

Example 41–14 Sample Back Up Script Properties - backup.properties

```
## backup.properties for backing up WebCenter Portal
## Specify valid values for your environment
############ User Input Required #############

##OPERATION - Specify either export or import
##  For backup scripts, specify OPERATION=export
##  For restore scripts, specify OPERATION=import
##
OPERATION=export

##Specify database information
##For backup scripts, specify source database details here
##
##   DATA_DIRECTORY    Location on the file system that contains the backup
##                     scripts files
##   DB_ORACLE_HOME    Database home directory
##   DB_ADMIN_USER     Database admin user
##   DB_ADMIN_PASSWORD Password for the database admin user
##   DB_SID            Database SID
##
DATA_DIRECTORY=/scratch/aime1/mywebcenterportal_scripts/mybackups
DB_ORACLE_HOME=/scratch/aime1/mywork/db1234
DB_ADMIN_USER=mydbadmin
DB_ADMIN_PASSWORD=mypassword
DB_SID=db1234

##Specify WebCenter Content vault and weblayout file location information
##For backup scripts, specify the source directories here
##
WCC_VAULT_LOC=/scratch/aime1/mywork/mymw/user_projects/domains/myDomainHome/ucm/cs
WCC_WEBLAYOUT_LOC=/scratch/aime1/mwork/mymw/user_projects/domains/myDomainHome/ucm
/cs

##Specify a connect string for each schema to export
##For backup scripts, specify connect strings for the source schemas here
## Use the format: <adminuser>/<password>@<serviceID>
## For example: mydbadmin/mypassword@db1234
##
DB_CONNECT_WEBCENTER_SCHEMA=mydbadmin/mypassword@db1234
DB_CONNECT_MDS_SCHEMA=mydbadmin/mypassword@db1234
DB_CONNECT_OCS_SCHEMA=mydbadmin/mypassword@db1234
DB_CONNECT_DISCUSSIONS_SCHEMA=mydbadmin/mypassword@db1234
DB_CONNECT_ACTIVITIES_SCHEMA=mydbadmin/mypassword@db1234
DB_CONNECT_PORTLET_SCHEMA=mydbadmin/mypassword@db1234

##Database schemas to export

##Identify source database schemas to export
##For back up scripts, specify source schema names here.
##
EXP_WEBCENTER_SCHEMA=myprefix_WEBCENTER
EXP_MDS_SCHEMA=myprefix_MDS
EXP_DISCUSSIONS_SCHEMA=myprefix_DISCUSSIONS
EXP_DISCUSSIONS_CRAWLER_SCHEMA=myprefix_DISCUSSIONS_CRAWLER
EXP_OCS_SCHEMA=myprefix_OCS
EXP_ACTIVITIES_SCHEMA=myprefix _ACTIVITIES
EXP_PORTLET_SCHEMA=myprefix_PORTLET
```

```
##Specify information for WLST export commands

##Specify general WLST information
##For backup scripts, specify details for the source system here
##
## WLST_ADMIN_USER     Name of the admin user connecting WLST to the Admin Server
## WLST_ADMIN_PASSWORD Password of the admin user
## WLST_ADMIN_CONSOLE  Host name and port of the Admin Server. Use the format:
##                     protocol://listen_address:listen_port
## WLST_LOCATION   Location of the WLST script. You must run WebCenter Portal WLST
##                 commands from your WebCenter Portal Oracle home directory
##                 (WCP_ORACLE_HOME/common/bin/wlst.sh)
## WLST_SCRIPT_LOCATION Location of the back up script (wlst_script.py)
## WCP_SERVER_NAME Name of the managed server on which the WebCenter Portal
##                 application (webcenter) is deployed
##
WLST_ADMIN_USER=mywlstadmin
WLST_ADMIN_PASSWORD=mypassword
WLST_ADMIN_CONSOLE=t3://myhost.com:24647
WLST_LOCATION=/scratch/aime1/mywork/mymw/mywcp/common/bin
WLST_SCRIPT_LOCATION=/scratch/aime1/mywebcenterportal_scripts/wlst_script.py
WCP_SERVER_NAME=WC_Spaces

## Specify information for security export
## (Policy store and credential store)
## Provide details about the security configuration file (jps-config.xml).
## For backup scripts, specify details about the source jps-config.xml here
##
## JPS_CONFIG_FILE              Location of the configuration file relative to
##                              the directory from which WLST commands run
## SRC_JPS_CONTEXT_POLICYSTORE  Name of a jps-context in the configuration file,
##                              where the source policy store is specified
## TGT_JPS_CONTEXT_POLICYSTORE  Name of another jps-context in the configuration
##                              file, where the target policy store is specified
## SRC_JPS_CONTEXT_CREDSTORE    Name of a jps-context in the configuration file,
##                              where the source credential store is specified
## TGT_JPS_CONTEXT_CREDSTORE  Name of another jps-context in the configuration
##                            file, where the target credential store is specified
## POLICYSTORE_FILE_NAME      Name and location of the policy store that you
##                            wamt to back up (as specified in JPS_CONFIG_FILE)
## CREDSTORE_FILE_NAME        Name and location of the credential store that you
##                            want to back up (location is as specified in
##                            JPS_CONFIG_FILE, with the file name cwallet.sso)
##
JPS_CONFIG_FILE=/scratch/aime1/mywork/mymw/user_projects/domains/MyDomainHome/conf
ig/fmwconfig/mybackup-jps-config.xml
SRC_JPS_CONTEXT_POLICYSTORE=mysourcePolicy
TGT_JPS_CONTEXT_POLICYSTORE=mytargetPolicy
SRC_JPS_CONTEXT_CREDSTORE=mysourceCred
TGT_JPS_CONTEXT_CREDSTORE=mytargetCred
POLICYSTORE_FILE_NAME=/scratch/aime1/mywebcenterportal_scripts/backup/backup-syste
m-jazn-data.xml
CREDSTORE_FILE_NAME=/scratch/aime1/mywebcenterportal_scripts/backup/cwallet.sso
```

Example 41–15 Sample Restore Script Properties - restore.properties

```
## restore.properties for restoring WebCenter Portal from a backup
## Specify valid values for your environment
############ User Input Required #############
```

```
##OPERATION - Specify either export or import
##  For backup scripts, specify OPERATION=export
##  For restore scripts, specify OPERATION=import
##
OPERATION=import

##Specify database information
##  For restore scripts, specify target database details here
##
##   DATA_DIRECTORY    Location on the file system that contains the backup
##                     files you want to restore
##   DB_ORACLE_HOME    Database home directory
##   DB_ADMIN_USER     Database admin user
##   DB_ADMIN_PASSWORD Password for the database admin user
##   DB_SID            Database SID
##
DATA_DIRECTORY=/scratch/aime1/mywebcenterportal_scripts/mybackups/2013.05.30-08.39
.28
DB_ORACLE_HOME=/scratch/aime1/mywork/db1234
DB_ADMIN_USER=mydbadmin
DB_ADMIN_PASSWORD=mypassword
DB_SID=db1234

##Specify WebCenter Content vault and weblayout file location information
##  For restore scripts, specify the target directories here
##
WCC_VAULT_LOC=/scratch/aime1/mywork/mymw/user_projects/domains/myDomainHome/ucm/cs
WCC_WEBLAYOUT_LOC=/scratch/aime1/mwork/mymw/user_projects/domains/myDomainHome/ucm
/cs

##Specify information for WLST import commands

##Specify general WLST information
## For restore scripts, specify details for the target system here
##
## WLST_ADMIN_USER     Name of the admin user connecting WLST to the Admin Server
## WLST_ADMIN_PASSWORD Password of the admin user
## WLST_ADMIN_CONSOLE  Host name and port of the Admin Server. Use the format:
##                     protocol://listen_address:listen_port
## WLST_LOCATION   Location of the WLST script. You must run WebCenter Portal WLST
##                 commands from your WebCenter Portal Oracle home directory
##                 (WCP_ORACLE_HOME/common/bin/wlst.sh)
## WLST_SCRIPT_LOCATION Location of the restore script (wlst_script.py)
## WCP_SERVER_NAME Name of the managed server on which the WebCenter Portal
##                 application (webcenter) is deployed
##
WLST_ADMIN_USER=mywlstadmin
WLST_ADMIN_PASSWORD=mypassword
WLST_ADMIN_CONSOLE=t3://myhost.com:24647
WLST_LOCATION=/scratch/aime1/mywork/mymw/mywcp/common/bin
WLST_SCRIPT_LOCATION=/scratch/aime1/mywebcenterportal_scripts/wlst_script.py
WCP_SERVER_NAME=WC_Spaces

## Specify information for security import
## (Policy store and credential store)
## Provide details about the security configuration file (jps-config.xml).
## For restore scripts, specify details about the target jps-config.xml here
##
## JPS_CONFIG_FILE             Location of the configuration file relative to
##                             the directory from which WLST commands run
```

```
## SRC_JPS_CONTEXT_POLICYSTORE  Name of a jps-context in the configuration file,
##                              where the source policy store is specified
## TGT_JPS_CONTEXT_POLICYSTORE  Name of another jps-context in the configuration
##                              file, where the target policy store is specified
## SRC_JPS_CONTEXT_CREDSTORE    Name of a jps-context in the configuration file,
##                              where the source credential store is specified
## TGT_JPS_CONTEXT_CREDSTORE  Name of another jps-context in the configuration
##                            file, where the target credential store is specified
## POLICYSTORE_FILE_NAME      Name and location of the policy store that you
##                            wamt to restore (as specified in JPS_CONFIG_FILE)
## CREDSTORE_FILE_NAME        Name and location of the credential store that you
##                            want to restore (location is as specified in
##                            JPS_CONFIG_FILE, with the file name cwallet.sso)
##
JPS_CONFIG_FILE=/scratch/aime1/mywork/mymw/user_projects/domains/MyDomainHome/conf
ig/fmwconfig/restore-jps-config.xml
SRC_JPS_CONTEXT_POLICYSTORE=mysourcePolicy
TGT_JPS_CONTEXT_POLICYSTORE=mytargetPolicy
SRC_JPS_CONTEXT_CREDSTORE=mysourceCred
TGT_JPS_CONTEXT_CREDSTORE=mytargetCred
POLICYSTORE_FILE_NAME=/scratch/aime1/mywebcenterportal_scripts/mybackups/2013.05.3
0-08.39.2/backup-system-jazn-data.xml
CREDSTORE_FILE_NAME=/scratch/aime1/mywebcenterportal_scripts/mybackups/2013.05.30-
08.39.28/cwallet.sso
```

41.8.2 Using Scripts to Back Up WebCenter Portal

This section describes how to set up, verify, and schedule WebCenter Portal backups using scripts files:

1. Create back up scripts (first time only).
2. Complete prerequisite tasks for security store back up (first time only).
3. Set back up parameters and customize scripts (first time only).
4. Run the back up script.
5. Verify back up archives.
6. Schedule regular back ups using the scripts.

Create back up scripts

(First time only)

1. Create a directory on the file system for your scripts and backups.

   For example: `/scratch/aime1/mywebcenterportal_scripts/backups`

2. Copy code from Example 41–12, "Sample Script - master_script.sh", paste into a text editor, and save the file as `master_script_backup.sh` into the directory you created in step 1.

   Note: Ensure that the script does not contain any hidden characters or DOS characters if running on Unix/Linux.

3. Copy code from Example 41–13, "Sample Script - wlst_script.py", paste into a text editor, and save the file as `wlst_script.py` in the same directory.

4. Copy code from Example 41–14, "Sample Back Up Script Properties - backup.properties", paste into a text editor, and save the file as `backup.properties` in the same directory.

Complete prerequisite tasks for security store back up

(First time only)

In the source environment:

1. Create a copy of your `jps-config.xml` file for the backup scripts.

   This file is located at:

   `SOURCE_DOMAIN_HOME/config/fmwconfig/jps-config.xml`

   Name the copy `mybackup-jps-config.xml` or similar and save it at the same location. For example, `/scratch/aime1/mywork/mymw/user_projects/domains/MyDomainHome/config/fmwconfig/mybackup-jps-config.xml`

2. Configure source and target information for backing up the *policy store* as follows:

   a. To point to the target policy store, add the following section (above the closing `</serviceInstances>` tag):

   ```
   <serviceInstance
           name="policystore.backup.xml"
           provider="policystore.xml.provider"
           location="<some_location>/mybackup-system-jazn-data.xml">
      <description>File Based Policy Store Service Instance</description>
    </serviceInstance>
   ```

   You can choose any location that the backup scripts can access. For example:

   `/scratch/aime1/mywebcenterportal_scripts/backups/backup-system-jazn-data.xml`

   Where, `backup-system-jazn-data.xml` is a copy of `system-jazn-data.xml` located at:

   `/scratch/aime1/mywork/mymw/user_projects/domains/MyDomainHome/config/fmwconfig/`

   b. Add and configure the following entries (above the closing `</jpsContexts>` tag):

   ```
   <jpsContext name="mysourcePolicy">
        <serviceInstanceRef ref="policystore.ldap"/>
   </jpsContext>

   <jpsContext name="mytargetPolicy">
        <serviceInstanceRef ref="policystore.backup.xml"/>
   </jpsContext>
   ```

3. Configure source and target information for backing up the *credential store* as follows:

   a. To point to the target credential store, add the following section (above the closing `</serviceInstances>` tag):

   ```
   <serviceInstance
           name="credstore.backup.xml"
           provider="credstore.xml.provider"
           location="<some_location>">
   ```

```
    <description>File Based Credential Store Service Instance</description>
  </serviceInstance>
```

You can choose any location that the backup scripts can access. For example, `/scratch/aime1/mywebcenterportal_scripts/backups`.

b. Add and configure the following entries (above the closing `</jpsContexts>` tag):

```
<jpsContext name="mysourceCred">
     <serviceInstanceRef ref="credstore.ldap"/>
</jpsContext>

<jpsContext name="mytargetCred">
     <serviceInstanceRef ref="credstore.backup.xml"/>
</jpsContext>
```

Set back up parameters and customize scripts

(First time only)

1. Open `backup.properties` in a text editor.
2. Ensure `OPERATION=export`.
3. Specify values for parameters in the file.

   Refer to Table 41–5 for a description of each parameter.

   Note: You can comment out parameters that are not required.

4. Customize the back up scripts, if required.

   To exclude objects, comment out the associated back up command code. To back up additional objects using the script, add the required code.

5. Save the changes.

Run the back up script

1. Set the following environment variables:

   `ORACLE_HOME`

   `ORACLE_SID`

   `TNS_ADMIN`

2. Verify that you have permissions to read and write to all directories used during the backup process.
3. Run the master back up script, specifying the name of the backup properties file and a log file name as follow:

   `sh` *`master_backup_script_name backup_properties_file_name log_file_name`*

   For example:

   ```
   sh master_script_backup.sh backup.properties mybackup.log
   ```

   The message "`Backup completed successfully...`" indicates when the backup process is complete and the directory in which your backups and the `export.log` file are located.

Each time you run the script, backup data is saved to a different folder under the main backup folder (DATA_DIRECTORY) so that previous backups are retained. Timestamp information is included in backup folder names so its easy to associate your backups with a particular date and time.

Verify back up archives

1. Navigate to the directory containing your data backups, that is, a timestamped folder under the location you specified for the DATA_DIRECTORY parameter in backup.properties.
2. Verify the following back up files are available:
   - one or more .dmp files
   - wcc_vault.tar
   - wcc_weblayout.tar
   - portletdata.ear
   - backup-system-jazn-data.xml
   - cwallet.sso
   - audit.xml
   - .log file

Schedule regular back ups using the scripts

Once you have verified your backup script configuration by successfully creating data backups with master_script_backup.sh, Oracle recommends that you schedule back ups at regular intervals.

Each time you run the script, backup data is saved to a different folder under the main backup folder (DATA_DIRECTORY) so that previous backups are retained.

To minimize data-integrity issue during data back up, Oracle recommends that you do not schedule backups during peak usage time.

41.8.3 Restoring WebCenter Portal from Backups Using Scripts

This section describes how to restore a WebCenter Portal installation from backups using scripts files:

1. Create restore scripts (first time only).
2. Complete prerequisite tasks for security store restore (first time only).
3. Set restore script parameters (first time only).
4. Restore database schemas manually.
5. Run the restoration script.
6. Verify restored data.

Create restore scripts

(First time only)

1. Duplicate the backup scripts that you created earlier master_script.sh wlst_script.py (following steps in section Using Scripts to Back Up WebCenter Portal) and copy them to a different location.

   For example: /scratch/aime1/mywebcenterportal_scripts/restore

2. Copy code from Example 41–15, "Sample Restore Script Properties - restore.properties", paste into a text editor, and save the file as `restore.properties` in the same directory.
3. Rename the files, if required.

   For example: `master_script_restore.sh`, `wlst_restore_script.py`, `restore.properties`

Restore database schemas manually

1. Ensure that all the target schemas were created using RCU and the names of the target schemas match the source schema names.
2. (Optional). If you want to point the default data sources to different schemas, use the WebLogic Server Admin Console to update the schema names, and database details.
3. Stop all the servers.
4. Restore schema data, as required.

   Important: Database schemas `WEBCENTER` and `MDS` *must* be restored together to ensure the data is in-sync.

   If you need to restore additional schemas, such as `PORTLET` or `OCS`, you must restore them at the same time, after `WEBCENTER` and `MDS`, and from the same point to maintain data integrity.

   This example shows you commands to restore `WEBCENTER` and `MDS` schemas:

```
./sqlplus "sys/password@serviceid as sysdba"
create or replace directory dmpdir as 'mydmpdirectory';
GRANT read,write ON directory dmpdir TO public;

##Drop WEBCENTER and MDS schemas ##

drop user srcprefix_WEBCENTER cascade;
drop user srcpreix_MDS cascade;
exit;
./impdp \"sys/password@serviceid as sysdba\" directory=dmpdir
dumpfile=webcenterportal.dmp SCHEMAS=srcprefix_WEBCENTER
./impdp \"sys/password@serviceid as sysdba\" directory=dmpdir dumpfile=mds.dmp
SCHEMAS=srcprefix_MDS
```

   Where:

   - `password` is the password for the system database user.
   - `serviceid` is the unique SID for the database. For example, `mydb1234`.
   - `directory` is the location on the database machine where the dump files are located.
   - `dumpfile` is the name of the file that contains data to be imported.
   - `SCHEMAS` identifies the target schemas. Schema names include the RCU suffix that was used during installation (`_WEBCENTER` and `_MDS`), along with a user supplied prefix. For example, `DEV_WEBCENTER`.

     Schema names on the source and target *must* match. For example, both schemas must be named `DEV_WEBCENTER`.

For example:

```
./sqlplus "sys/mypassword@db1234 as sysdba"
create or replace directory dmpdir as
'/scratch/mywebcenterportal_scripts/backup/2013.05.04-02.36.48';
GRANT read,write ON directory dmpdir TO public;

##Drop WEBCENTER and MDS schemas ##

drop user DEV_WEBCENTER cascade;
drop user DEV_MDS cascade;
exit;
./impdp \"sys/mypassword@db1234 as sysdba\" directory=dmpdir dumpfile=wcdmp.dmp
SCHEMAS=DEV_WEBCENTER
./impdp \"sys/mypassword@db1234 as sysdba\" directory=dmpdir
dumpfile=mdsdmp.dmp SCHEMAS=DEV_MDS
```

Note: If you need to restore other schemas, such as `DISCUSSIONS`, `PORTLETS`, `ACTIVITIES`, and `OCS`, then do so now before starting the servers.

5. Start all the servers.

Complete prerequisite tasks for security store restore

(First time only)

In the target environment:

1. Create a copy of your `jps-config.xml` file for the restore scripts.

   This file is located at:

   `TARGET_DOMAIN_HOME/config/fmwconfig/jps-config.xml`

   Name the copy `myrestore-jps-config.xml` or similar and save it at the same location. For example, `/scratch/aime1/mywork/mymw/user_projects/domains/MyDomainHome/config/fmwconfig/myrestore-jps-config.xml`

2. Configure source and target information for restoring the *policy store* as follows:

   a. To point to the source policy store, add the following section (above the closing `</serviceInstances>` tag):

   ```
   <serviceInstance
           name="policystore.backup.xml"
           provider="policystore.xml.provider"
           location="<some_location>/mybackup-system-jazn-data.xml">
      <description>File Based Policy Store Service Instance</description>
    </serviceInstance>
   ```

   The location you specify must contain a previously backed up policy store that you want to restore. For example, `/scratch/aime1/mywebcenterportal_scripts/backups/2013.06.19-09.20.14/backup-system-jazn-data.xml`

   b. Configure the following entries (above the closing `</jpsContexts>` tag):

   ```
   <jpsContext name="targetPolicy">
        <serviceInstanceRef ref="policystore.ldap"/>
   </jpsContext>
   ```

```
<jpsContext name="sourcePolicy">
     <serviceInstanceRef ref="policystore.backup.xml"/>
</jpsContext>
```

3. Configure source and target information for restoring the *credential store* as follows:

   a. To point to the source credential store, add the following section (above the closing `</serviceInstances>` tag):

   ```
   <serviceInstance
          name="credstore.backup.xml"
          provider="credstore.xml.provider"
          location="<some_location>">
     <description>File Based Credential Store Service Instance</description>
    </serviceInstance>
   ```

   The location you specify must contain a previously backed up credential store (`cwallet.sso`) that you want to restore. For example, `/scratch/aime1/mywebcenterportal_scripts/backups/2013.06.19-09.20.14.`

   b. Configure the following entries (above the closing `</jpsContexts>` tag):

   ```
   <jpsContext name="targetCred">
        <serviceInstanceRef ref="credstore.ldap"/>
   </jpsContext>

   <jpsContext name="sourceCred">
        <serviceInstanceRef ref="credstore.backup.xml"/>
   </jpsContext>
   ```

Set restore script parameters

(First time only)

1. Open `restore.properties` in a text editor.
2. Ensure that `OPERATION=import`.
3. Specify values for all parameters in the file.

   Refer to Table 41–5 for a description of each parameter.
4. Save the changes.

Run the restoration script

1. Set the following environment variables:

   `ORACLE_HOME`

   `ORACLE_SID`

   `TNS_ADMIN`
2. Verify that you have permissions to read and write to all directories used during the restore process.
3. Run the master restoration script, specifying the name of the restore properties file and a log file name as follow:

   `sh` *`master_restore_script_name bestore_properties_file_name log_file_name`*

   For example:

```
sh master_script_restore.sh backup.properties myrestore.log
```

The message "`Restoration completed successfully...`" indicates when the restore process is complete and the directory where the `restore.log` file is located.

Verify restored data

Check your WebCenter Portal installation:

1. If you import one or more database schemas, shut down and restart those databases, and restart all managed servers.
2. Verify the target WebCenter Portal instance includes the restored data.

41.9 Cloning a WebCenter Portal Environment

Cloning creates a new WebCenter Portal environment based on existing ones. You can install, configure, customize, and validate your WebCenter Portal installation and when the system is stable, create another environment by copying all the components and their configurations from the source environment. This saves time as you do not need to redo all the changes you incorporated and tested in the source environment. For more information, see the "Moving Oracle WebCenter Portal to a Target Environment" section in *Oracle Fusion Middleware Administrator's Guide*.

Part IX

Lifecycle: Portal Framework Applications

This part of *Oracle Fusion Middleware Administering Oracle WebCenter Portal* presents Portal Framework application lifecycle operations.

Part IX contains the following chapters:

- Chapter 42, "Deploying Portal Framework Applications"
- Chapter 43, "Administering Portal Framework Applications Using the Administration Console"
- Chapter 44, "Managing Export, Import, Backup, and Recovery for Portal Framework Applications"

Many different people participate in the Portal Framework application life cycle. This guide only describes life cycle tasks that are performed by *administrators* whose main responsibility is to deploy, configure, and manage portal applications. For an overview of the tools and techniques for managing a Portal Framework application throughout its entire life cycle, see the "Understanding the Portal Framework Application Life Cycle" chapter in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

42

Deploying Portal Framework Applications

This chapter provides instructions for deploying, undeploying, and redeploying Portal Framework applications from an Enterprise Archive, or EAR file, created with Oracle JDeveloper.

- Section 42.1, "Deploying Portal Framework Applications"
- Section 42.2, "Undeploying Portal Framework Applications"
- Section 42.3, "Redeploying Portal Framework Applications"
- Section 42.4, "Post-Deployment Configuration"

For information on how to create an EAR file, see the "Deploying a Portal Framework Application to a WebLogic Managed Server" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

This chapter does not contain instructions for deploying or installing Oracle WebCenter Portal's out-of-the box portal application "*WebCenter Portal*". For information about installing WebCenter Portal and its related components, see the "Installing Oracle WebCenter Portal" chapter in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*. For information about deploying WSRP and PDK-Java portlet producer applications, see Section 21.11, "Deploying Portlet Producer Applications."

Permissions: To perform the tasks in this chapter, you must be granted the following roles:

- **WebLogic Server:** `Admin` or `Deployer` role granted through the Oracle WebLogic Server Administration Console.
- **Portal Framework application**: `Administrator` role granted through the Administration Console.

See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

42.1 Deploying Portal Framework Applications

This section describes the steps required to deploy a Portal Framework application created in JDeveloper to a production domain. The deployment steps in this section assume that you are deploying an `EAR` file, know its location, and that the domain to which you want to deploy exists.

For information on how to create a WebLogic Server domain, see the "Creating a New Domain" section in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter*

Portal. For more information about deploying applications, see *Oracle Fusion Middleware Deploying Applications to Oracle WebLogic Server*.

This section includes the following topics:

42.1.1 Deployment Roadmap

The flow chart and table in this section provide an overview of the prerequisites and tasks required to deploy a Portal Framework application to an Oracle WebLogic Managed Server. Figure 42–1 shows the steps to deploy a Portal Framework application, and the roles that will carry them out.

Figure 42–1 Deploying a Portal Framework Application to a Managed Server

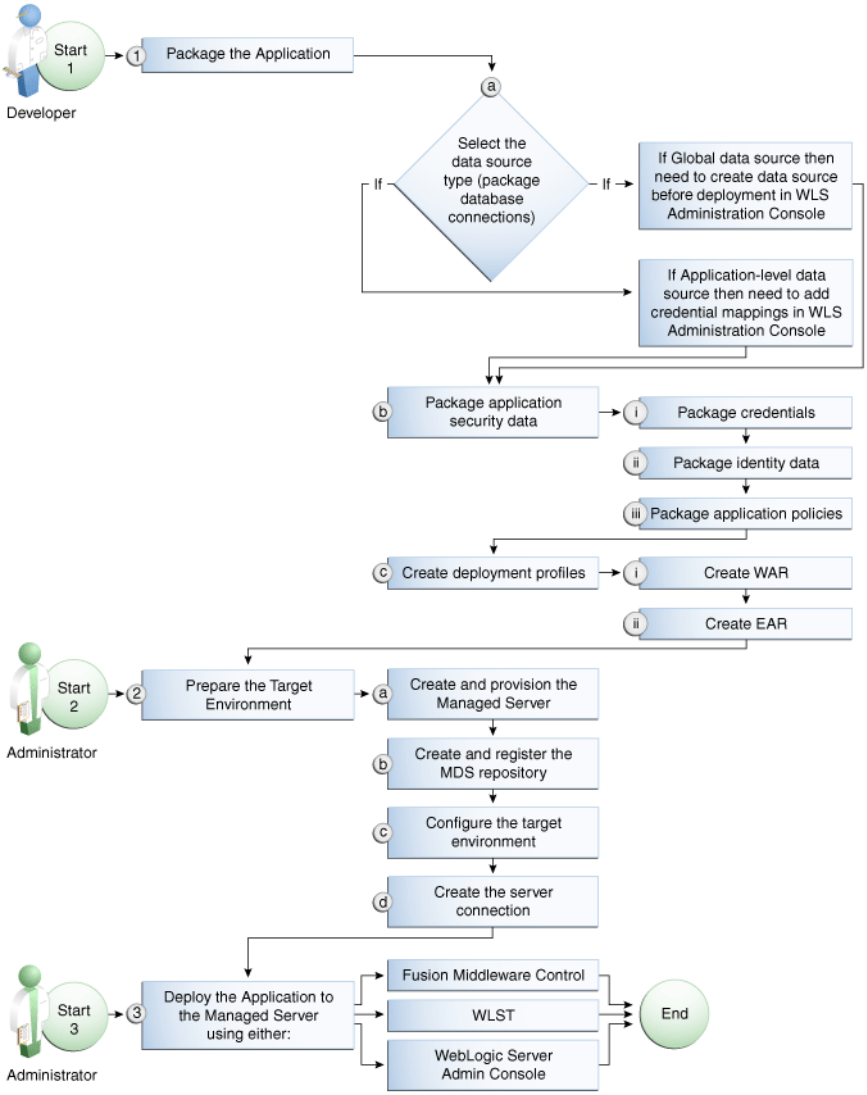


Table 42–1 shows the tasks, sub-tasks and who will need to carry them out to deploy a Portal Framework application from JDeveloper.

Table 42–1 Deploying a Portal Framework Application to a Managed Server

Actor	Task	Sub-task	Notes
Developer	**1.** Package the Application	**1.a** Select the data source type (package database connections)	You can use either a global data source or an application-level data source. If using a global data source, then you need to create the data source in the WLS Administration Console before deploying. If using an application-level data source, then you need to add credential mappings in the WLS Administration Console after deploying.
		1.b Package application security data	This sub-task consists of packaging the credentials, identity data, and application policies.
		1.c Create deployment profiles	This sub-task consists of creating the WAR and EAR files.
Administrator	**2.** Prepare the Target Environment	**2.a** Create and provision the Managed Server	
		2.b Create and register the MDS repository	
		2.c Configure the target environment	
		2.d Create the server connection	
Administrator	**3.** Deploy the Application to a Managed Server		The final step is to deploy the application to the Managed Server using either Fusion Middleware Control, WLST, or the WLS Admin Console.

42.1.2 Deployment Prerequisites

You can deploy Portal Framework applications to any WebLogic Managed Server instance that is provisioned with the Oracle WebCenter Portal libraries.

> **Note:** Oracle does not recommend deploying Portal Framework applications to any of the preconfigured Managed Servers created during the installation, or to the Administration Server. For Portal Framework applications, follow the process described in the "Extending an Existing Domain" section in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal* to create and provision a new WLS Managed Server before deploying applications. For portlet producer applications, you can optionally create a new WebLogic Managed Server or deploy to the `WC_Portlet` server.

Before deploying, you must:

- Prepare the application EAR file, as described in Section 42.1.3, "Preparing the Application EAR File."
- Create a WebLogic Managed Server, as described in Section 42.1.4, "Creating a Managed Server."

- Create and register a Metadata Service (MDS) repository, as described in Section 42.1.5, "Creating and Registering the Metadata Service Repository."

> **Note:** You must delete runtime customizations (customizations not done through JDeveloper) before deploying an updated application that has had major changes to artifacts such as pages, connections, or task flows.

After completing these steps, continue by deploying the application as described in Section 42.1.6, "Deploying the Application to a WebLogic Managed Server."

42.1.3 Preparing the Application EAR File

Before you deploy an application, you must first create a deployment profile. The deployment profile packages the Portal Framework application and its associated files so that the application can be deployed to an Oracle WebLogic Managed Server as an EAR file.

For information on how to create a deployment profile (and the resulting EAR file) for an application, see the "Packaging a Framework Application" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

42.1.3.1 EAR File Contents

The EAR file packages multiple information artifacts, which include:

- The application itself – the various pieces of the application such as `.jspx`, `.jar`, and `.class` files.
- Application Configuration – which contains the URL endpoints and properties of connections to tools and services and producers that are configured for this application.
- Application Metadata – which is an export of the application metadata created during the design time of the application.
- Portlet Customizations – which contain customization settings and data for portlets. This information is maintained within the producer, but is exported when an application with registered producers is packaged. This customization data is packaged with the rest of the metadata of a Portal Framework application.

42.1.4 Creating a Managed Server

Before deploying a Portal Framework application, you must create a WebLogic Managed Server based on the "Oracle WebCenter Portal Framework" template that contains all the required shared libraries and the MDS Repository. If a Portal Framework application has been portletized it should be deployed to the Oracle WebCenter Portal Custom Services Producer server (`WC_CustomServicesProducer`). A portletized application cannot be deployed to the Oracle WebCenter Portal Custom Portal server as it lacks the required portlet libraries. Note that the Oracle WebCenter Portal Custom Services Producer and Oracle WebCenter Portal Custom Portal servers have not only the MDS schema targeted to them but also WebCenter Portal and Activities.

For instructions on how to create a new managed server, see the "Extending an Existing Domain" section in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*. For instructions on how to create a new domain, see the "Creating a

New Domain" section in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*.

42.1.5 Creating and Registering the Metadata Service Repository

Before deploying an application to a Managed Server, you may need to create and register a Metadata Service (MDS) repository schema for the application on the WebLogic Domain's Administration Server instance. The target server (Oracle WebCenter Portal Custom Portal server or Oracle WebCenter Portal Custom Services Producer server) already has the MDS data source configured, so this step is only required if you do not want to use the pre-configured server MDS data source. Do not, however, create a new MDS schema if it is being shared by other applications.

Caution: If you deploy using the MDS schema that was created during the WebCenter Portal installation instead of using a custom schema as described in this section, you risk damaging data in those schemas.

At deployment time, some configuration information and application metadata exported into the EAR file must be imported into the MDS schema for use in the production environment. Importing the metadata occurs automatically during deployment when you select a target metadata schema (as explained in Section 42.1.6, "Deploying the Application to a WebLogic Managed Server").

You create the MDS schema using the Repository Creation Utility (RCU). After creating the MDS schema, you must register it using either Fusion Middleware Control, or from the command line using WLST.

This section contains the following subsections:

- Section 42.1.5.1, "Creating the MDS Schema Using the Repository Creation Utility"
- Section 42.1.5.2, "Registering the MDS Schema Using Fusion Middleware Control"
- Section 42.1.5.3, "Registering the MDS Schema Using WLST"

42.1.5.1 Creating the MDS Schema Using the Repository Creation Utility

Before you deploy an application, you must first create the MDS schema on a database server instance using the Repository Creation Utility (RCU), and then register it on the administration server for the domain to which you're deploying so that the application's metadata can also be deployed.

When following these instructions, be sure to note the MDS schema name and the login credentials for accessing it. You need this information for subsequent steps in the deployment process.

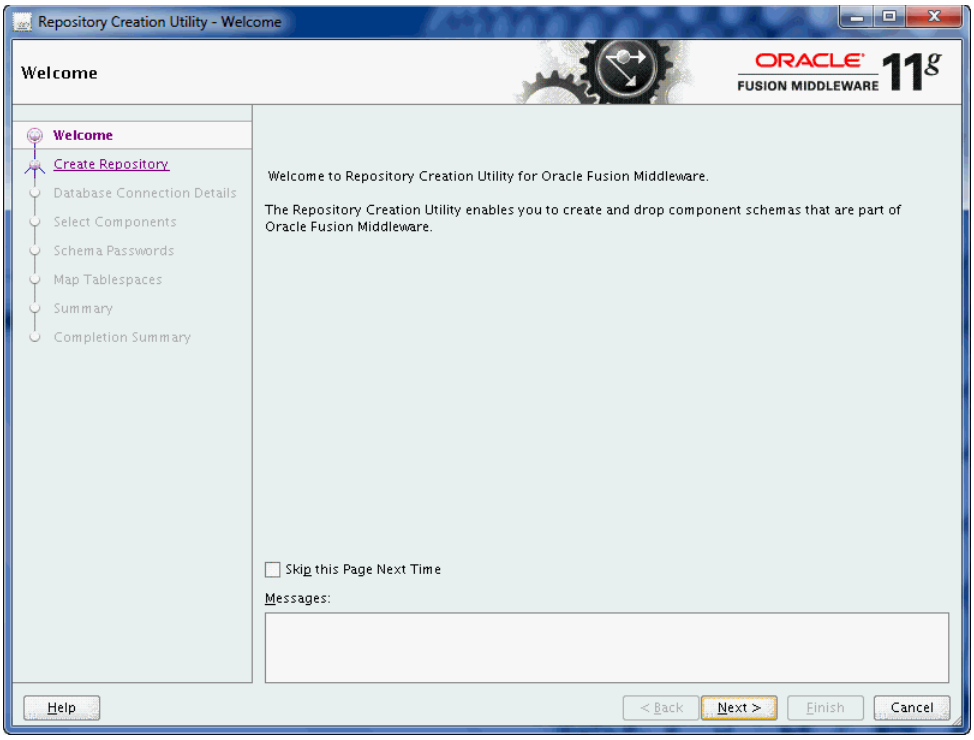
To create the MDS schema:

1. Navigate to *`RCU_HOME`*`/bin` and start the RCU with the following command:

   ```
   rcu
   ```

   The RCU Welcome page displays (see Figure 42–2).

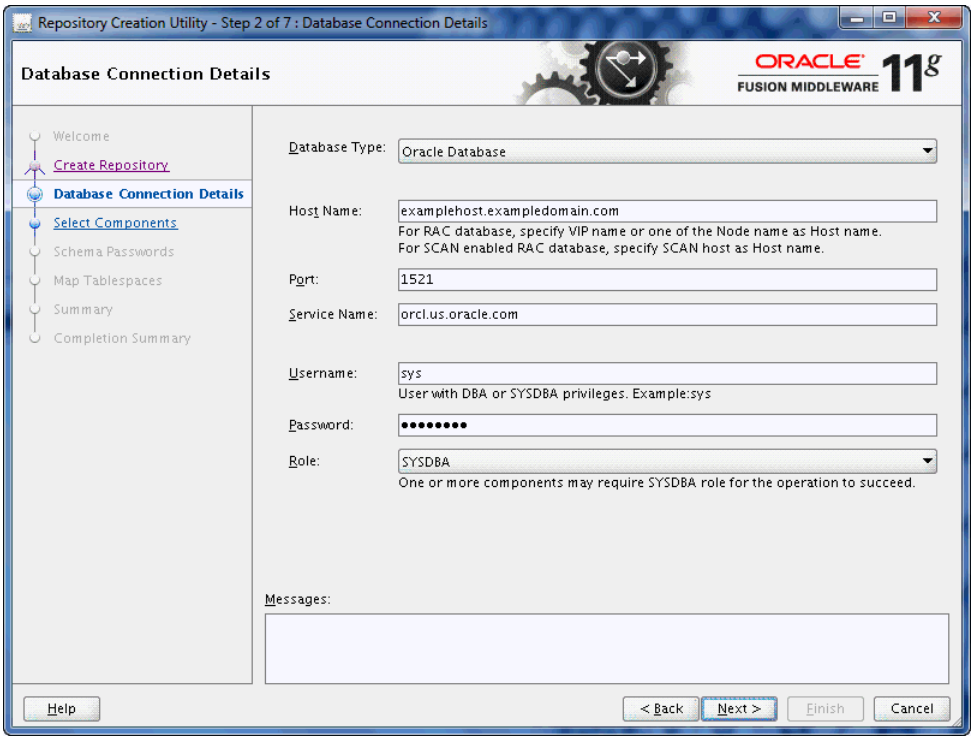
Figure 42–2 RCU Welcome Page



2. Click **Next**.
3. On the Create Repository page, select **Create**, and then click **Next**.

   The Database Connection Details page displays (see Figure 42–3).

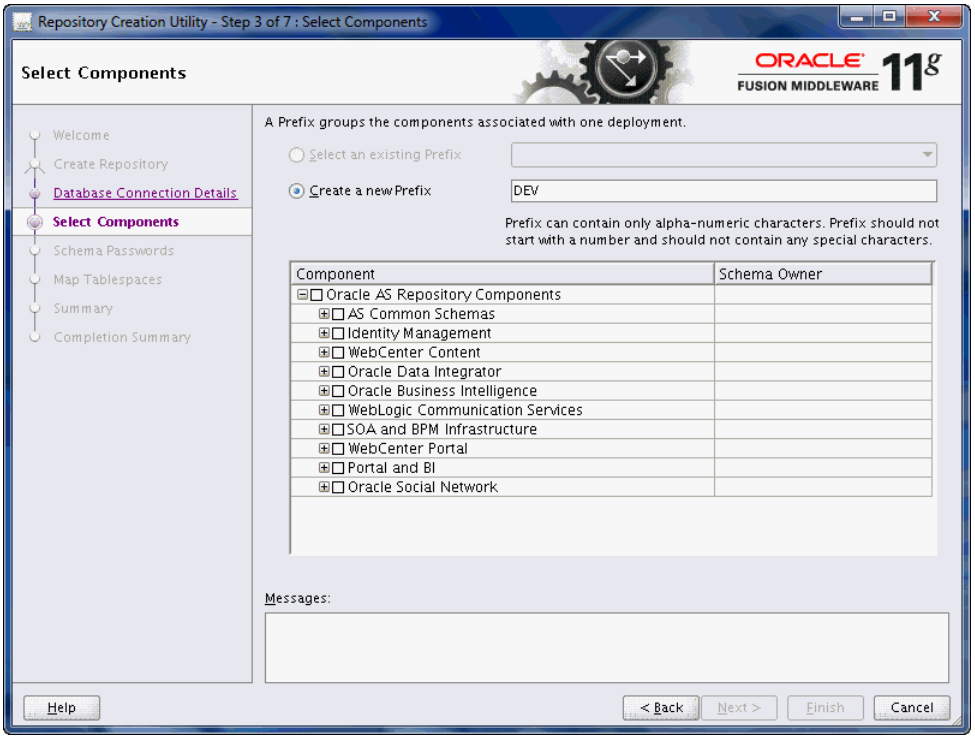
Figure 42–3 Database Connection Details Page



4. Provide the connection details for the database to which to add the schema by selecting the **Database Type**, entering the **Host Name**, **Port**, **Service Name**, **Username** and **Password** and clicking **Next**.
5. Click **OK** when prompted by the Prerequisites pop-up.

   The Select Components page displays (see Figure 42–4).

Figure 42–4 Select Components Page



6. Check **Create a New Prefix** and enter a prefix to be prepended to the schema name.
7. Check the **Metadata Services** component under AS Common Schemas. All other components should be left unchecked.
8. Click **Next**, and click **OK** when prompted by the Prerequisites pop-up.

   The Schema Passwords page displays.
9. Select how the schema password should be applied, and enter and confirm the password.
10. Click **Next**.
11. On the Map Tablespaces page, click **Next**
12. When prompted to create the tablespaces, click **OK**, and then click **OK** again when the operation is complete.
13. On the Summary page, click **Create** to create the schema.
14. On the Completion Summary page that indicates the successful completion of creating the schema, click **Close**.

42.1.5.2 Registering the MDS Schema Using Fusion Middleware Control

Before you deploy your application, you must first register the new MDS schema with the domain so that applications running on the Managed Server can access it.

To register the MDS repository using Fusion Middleware Control:

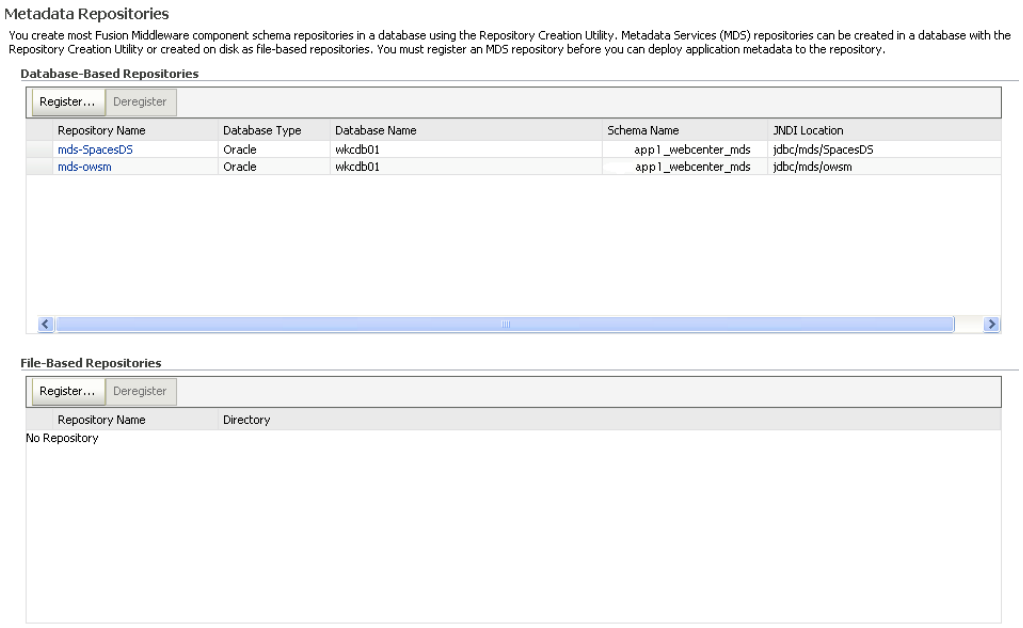
1. Open Fusion Middleware Control and log in to the target domain.

   For information on logging into Fusion Middleware Control, see Section 6, "Starting Enterprise Manager Fusion Middleware Control."

2. In the Navigation pane, expand the **farm**, then **WebLogic Domain**.
3. Select the domain to which you want to deploy.
4. From the WebLogic Domain menu, select **Metadata Repositories**.

   The Metadata Repositories page displays (see Figure 42–5).

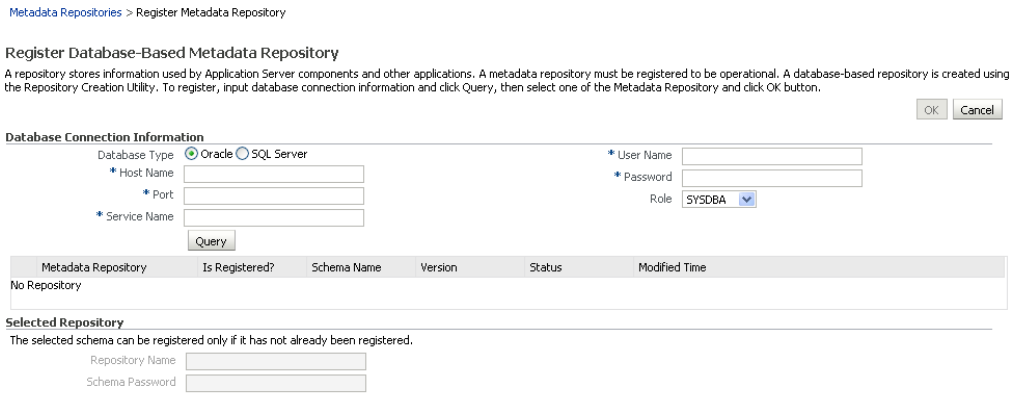
Figure 42–5 Metadata Repositories Page



5. In the Database-Based Repositories section, click **Register**.

   The Register Database-Based Metadata Repository page displays (see Figure 42–6).

Figure 42–6 Register Database-based Metadata Repository Page



6. In the Database Connection section, enter the following information:
   - **Database** - select the type of database.
   - **Host Name** - enter the name of the host.
   - **Port** - enter the port number for the database (for example, `1521`).
   - **Service Name** - enter the service name for the database. The default service name for a database is the global database name, comprising the database name, such as `orcl`, and the domain name, such as `example.com`. In this case, the service name would be `orcl.example.com`.
   - **User Name** - enter a username for the database which is assigned the SYSDBA role (for example, `SYS`).
   - **Password** - enter the password for the user.
   - **Role** - select a database role (for example, **SYSDBA**).
7. Click **Query**.

   A table is displayed that lists the schemas and their metadata repositories that are available in the database.
8. Select a repository, then enter the following information:
   - **Repository Name** - enter a name for the MDS schema.
   - **Schema Password** - enter the schema password you specified when you created the schema.
9. Click **OK**.

   The repository is registered with the Oracle WebLogic Server domain.

42.1.5.3 Registering the MDS Schema Using WLST

You can also use WLST to register a database-based MDS repository from the command line using the `registerMetadataDBRepository` command.

To register the MDS schema using WLST:

1. Start WLST as described in Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."
2. Register the MDS schema using the following command:

```
registerMetadataDBRepository(name='mds_name', dbVendor='db_vendor',
```

```
host='host_name', port='port_number',
dbName='db_name', user='username', password='password',
targetServers='target_server')
```

Where:

- *mds_name* is the name of the MDS schema to register.
- *db_vendor* is the vendor of the database being used.
- *host_name* is the fully qualified server name of the database server.
- *port_number* is the port number of the database server.
- *db_name* is the name of the database being used to store the MDS.
- *username* is the database schema user name.
- *password* is the database schema password.
- *target_server* is the name of the target server. For multiple targets, separate the target server names with a comma. Be sure to include the WLS administration server in the list of targets so that the MDS database repository name appears in the Deployment Plan dialog when you deploy your application to it.

For example, to register the MDS schema `mds1` on the Oracle database `orcl` on the target server `server1` with the host name `example.com`, you would use the following command:

```
registerMetadataDBRepository(name='mds1', dbVendor='ORACLE',
host='example.com',
port='1521',dbName='orc1', user='username', password='password',
targetServers='server1','AdminServer')
```

42.1.6 Deploying the Application to a WebLogic Managed Server

Table 42–2 lists the Managed Servers to which you can deploy your various applications.

For Portal Framework applications created in JDeveloper, follow the process described in the "Extending an Existing Domain" section in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal* to create and provision a new Oracle WebCenter Portal Custom Portal server, or if portletized, Oracle WebCenter Portal Custom Services Producer server before deploying.

Table 42–2 Deployment Targets

Application Type	Managed Server Name
Portal Framework applications	`WC_CustomPortal`
WebCenter Portal Portlet Producer applications	`WC_CustomServicesProducer`
Non-WebCenter Portal Portlet Producer applications	`WC_Portlet` For portlet producer applications, you can either create a Managed Server instance or deploy to the `WC_Portlet` server.

Note: Oracle does not recommend deploying Portal Framework applications to any of the preconfigured Managed Servers created during the installation, or to the Administration Server.

Portal Framework applications can be deployed in several ways as described in the following sections:

- Section 42.1.6.1, "Choosing the Information Artifact Store"
- Section 42.1.6.2, "Choosing the Data Source"
- Section 42.1.6.3, "Deploying Applications Using Oracle JDeveloper"
- Section 42.1.6.4, "Deploying Applications Using Fusion Middleware Control"
- Section 42.1.6.5, "Deploying Applications Using WLST"
- Section 42.1.6.6, "Deploying Applications Using the WLS Administration Console"
- Section 42.1.6.7, "Saving and Reusing the Deployment Plan"

42.1.6.1 Choosing the Information Artifact Store

As explained in Section 42.1.3, "Preparing the Application EAR File," the packaged EAR file consists of several information artifacts, which includes application binaries, application configuration, application metadata, and portlet customizations.

During the deployment, these information artifacts must be moved to the right information store in the instance where application is deployed. The target information stores for these artifacts are as described in Table 42–3:

Table 42–3 Information Artifact Target Stores

Information Artifact	Target Information Store
Application Binaries	Target Server Instance
Application Configuration	MDS
Application Metadata	MDS
Portlet Customizations	Target Producer

Regardless of the tool you choose to deploy, you must supply the target information store locations for correct deployment. The application deployment fails if the MDS location is incorrect or not supplied. The application will still deploy, however, if the target producer is incorrectly specified. If you incorrectly specify the target producer, the portlets are not imported automatically and, consequently, are not operational. If that happens, do one of the following:

- Edit the portlet producers connections post-deployment using Fusion Middleware Control (see Section 21.2.1, "Registering a WSRP Producer Using Fusion Middleware Control" and Section 21.4.1, "Registering an Oracle PDK-Java Producer Using Fusion Middleware Control") or WLST commands (see Section 21.2.2, "Registering a WSRP Producer Using WLST" or Section 21.4.2, "Registering an Oracle PDK-Java Producer Using WLST"), and redeploy the application.
- Export and import the portlet customization using WLST commands (see Section 44.1, "Exporting and Importing Portal Framework Applications for Data Migration").

Note: If the application is deployed and the target producer is incorrectly specified but the target exists, the portlets are imported but to the wrong producer and the portlets are not operational.

42.1.6.2 Choosing the Data Source

There are three basic options for data sources:

- Deploying to Oracle WebCenter Portal's Custom Portal Managed Server using pre-existing data sources
- Deploying to a Managed Server using global data sources not named WebCenterDS or ActivitiesDS
- Deploying to a Managed Server using local application context data sources of any name

This section describes the benefits and drawbacks of each of these options:

Deploying to Oracle WebCenter Portal's Custom Portal Managed Server using pre-existing data sources

This option requires the least effort in enabling a Portal Framework application to access the required data sources, and is the recommended deployment path.

To deploy using pre-existing data sources, deselect the **Auto Generate and Synchronize weblogic-jdbc.xml Descriptors During Deployment** check box on the Application Properties Deployment screen in JDeveloper.

If your application has existing database connections configured for WebCenterDS and/or ActivitiesDS and these are not named "webcenter/CustomPortal" and "activities/CustomPortal" respectively, either the database connections should be deleted from the application prior to deployment, or the database connections should be created and named following the naming convention.

Deploying to a Managed Server using global data sources not named WebCenterDS or ActivitiesDS

Use this deployment path when the application is not intended to run on a Managed Server created with the Oracle WebCenter Custom Portal template, or is intended to run against custom data sources not named "WebCenterDS" or "ActivitiesDS".

For this option the Portal Framework application should have had database connections created and associated as either the `WEBCENTER` or `ACTIVITIES` schema. The **Auto Generate and Synchronize weblogic-jdbc.xml Descriptors During Deployment** check box on the Application Properties Deployment screen in JDeveloper should be deselected. The global data sources intended to be used on the WLS server requires them to be created with the JNDI names matching those of the database connections created for the application in the JDeveloper project. For more information, see the "Creating a JDBC Data Source" section in *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server*.

Deploying to a Managed Server using local application context data sources of any name

Use this deployment path if the application local context data sources are sufficient.

This choice requires only that the Portal Framework application has a database connection created for and associated with `WebCenterDS` and/or `ActivitiesDS` (depending on which tools and services are being used in the application). The

Application Properties Deployment screen in JDeveloper should have the **Auto Generate and Synchronize weblogic-jdbc.xml Descriptors During Deployment** check box selected.

42.1.6.3 Deploying Applications Using Oracle JDeveloper

You can deploy Portal Framework applications to a WebLogic Server instance directly from a development environment using Oracle JDeveloper, if you have the necessary credentials to access the WebLogic Server. For more information, see the "Creating a WebLogic Managed Server Connection" and "Deploying a Portal Framework Application to a Managed Server" sections in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

42.1.6.4 Deploying Applications Using Fusion Middleware Control

When deploying a Portal Framework application using Fusion Middleware Control you must know the location of the application archive, and whether a deployment plan exists for the application. For more information about deployment plans, see Section 42.1.6.7, "Saving and Reusing the Deployment Plan."

> **Note:** Metadata repository and ADF connection details specified during deployment are not stored as part of the deployment plan. You will need to specify these deployment properties each time you deploy the application.
>
> If you plan to update and deploy the application frequently and want to maintain these configuration changes, it is recommended that you make those configuration changes post-deployment using WLST or Fusion Middleware Control. Such configuration changes are saved in the deployment plan and persisted in the MDS repository and do not need to be set again when you redeploy the application.

To deploy a Portal Framework application using Fusion Middleware Control:

1. Log in to Fusion Middleware Control.

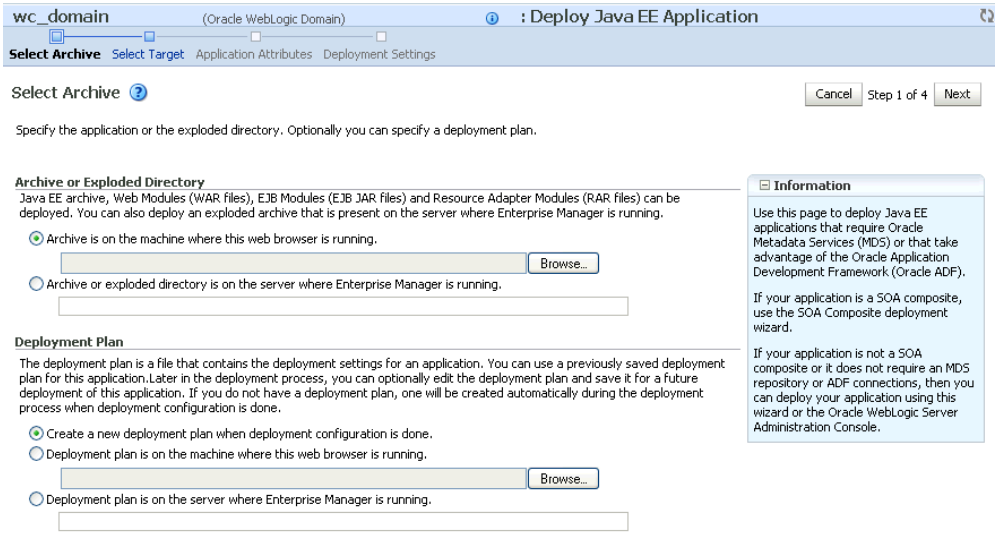
   See Section 6.1, "Displaying Fusion Middleware Control Console."

2. In the Navigation pane, expand **WebLogic Domain** and click the domain in which your target Managed Server was created.

3. From the WebLogic Domain menu, select **Application Deployment** > **Deploy**.

   The Select Archive page displays (see Figure 42–7).

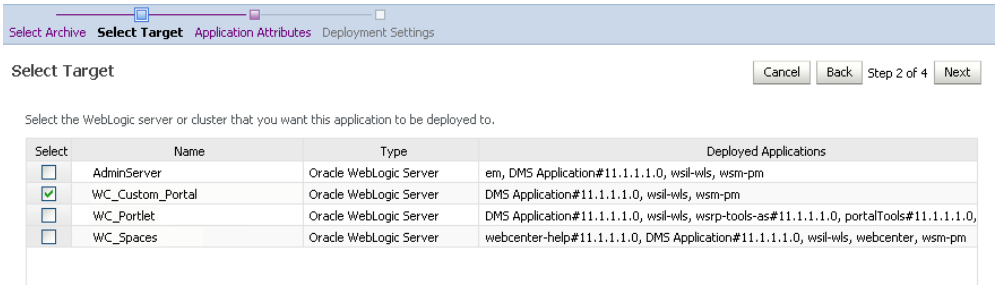
Figure 42–7 Select Archive Page



4. In the Archive or Exploded Directory section, do one of the following:
   - Select **Archive is on the machine where this web browser is running** and enter the location of the archive or click **Browse** to find the archive file.
   - Select **Archive or exploded directory is on the server where Enterprise Manager is running** and enter the location of the archive or click **Browse** to find the archive file.
5. In the Deployment Plan section, do one of the following:
   - Select **Create a new deployment plan when deployment configuration is done** to automatically create a new deployment plan after the redeployment process.
   - Select **Deployment plan is on the machine where this web browser is running** and enter the path to the plan or click **Browse** to find the plan.
   - Select **Deployment plan is on the server where Enterprise Manager is running** and enter the path to the plan or click **Browse** to find the plan.
6. Click **Next**.

   The Select Target page displays (see Figure 42–8).

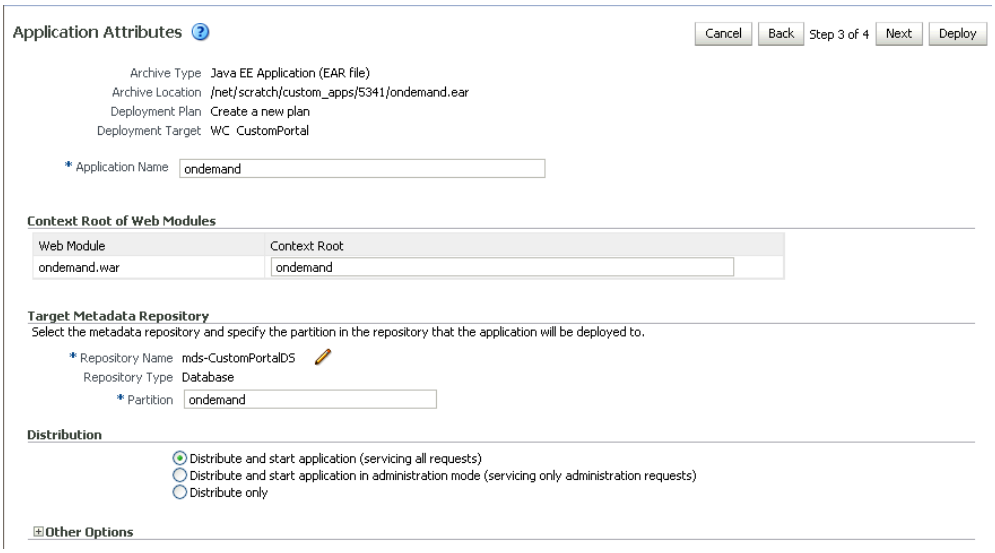
Figure 42–8 Select Target Page



7. Select the target server(s) to deploy the application and click **Next**.

   The Application Attributes page displays (see Figure 42–9).

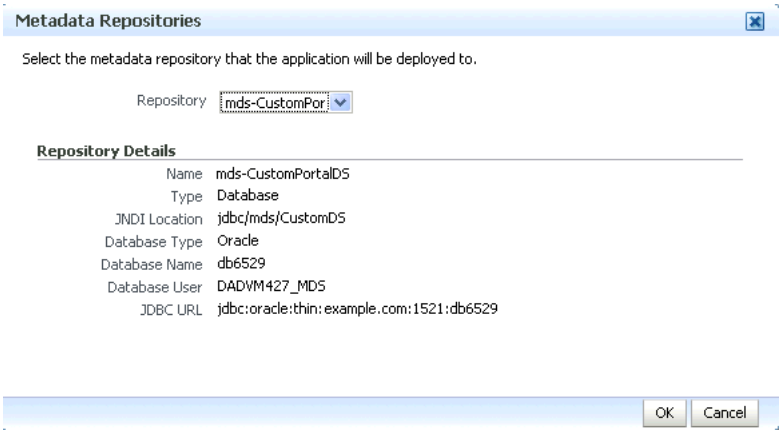
Figure 42–9 Application Attributes Page



8. Under Target Metadata Repository, click the icon to display the Metadata Repositories window, from where you can select the repository for the application, as shown in Figure 42–10. Use the Repository drop-down list to select the required repository and then click **OK**.

> **Note:** The Target Metadata Repository option only displays if the application has metadata to be imported into the MDS repository. This option does not display for a portlet producer application.

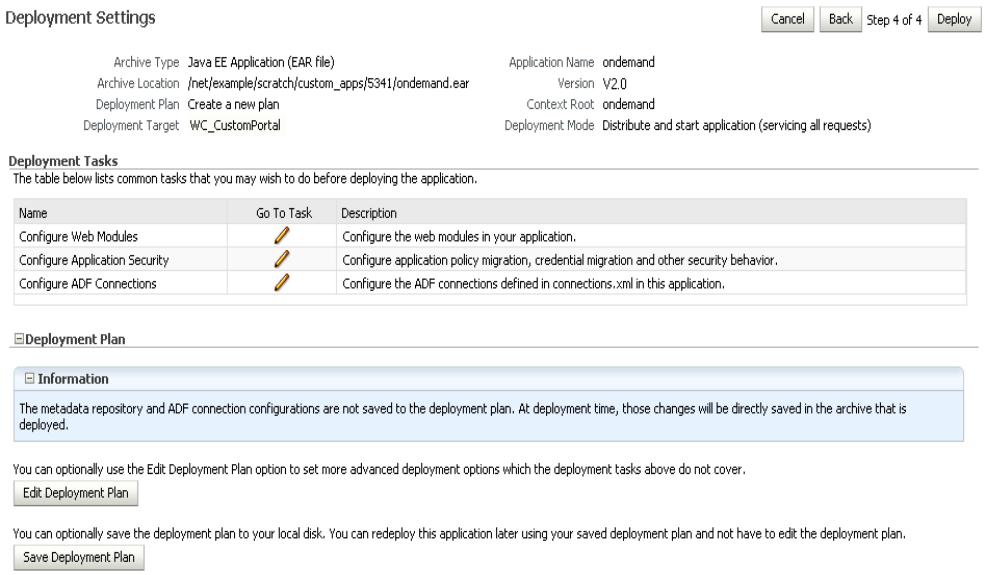
Figure 42–10 Select Metadata Repository Window



9. Enter the name of the partition to use in the repository (typically, the name of the application). Each application must have a unique partition in the repository.
10. Click **Next**.

    The Deployment Settings page displays (see Figure 42–11).

Figure 42–11 Deployment Settings Page

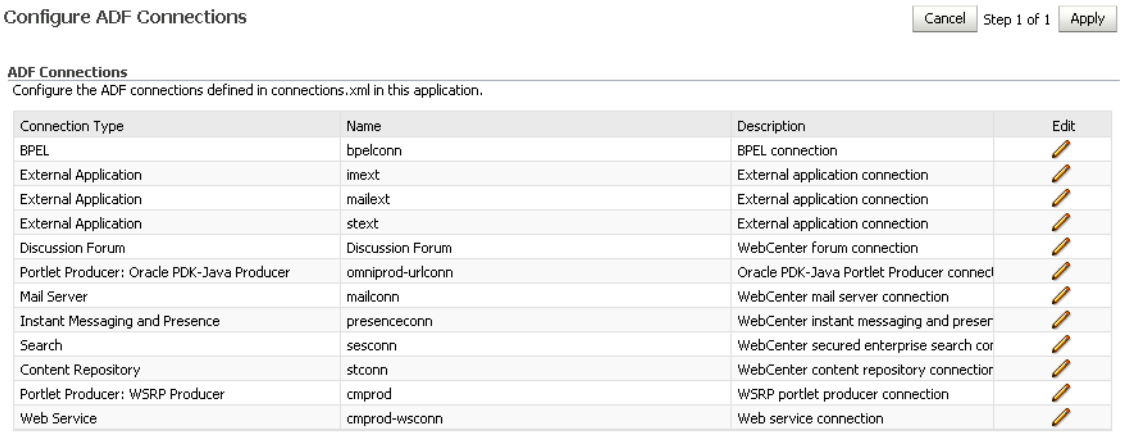


You have now provided the Target MDS location (described in Section 42.1.6, "Deploying the Application to a WebLogic Managed Server").

11. Click the **edit** icon for Configure ADF Connections to check connection settings associated with the Portal Framework application.

    The Configure ADF Connections page displays (see Figure 42–12).

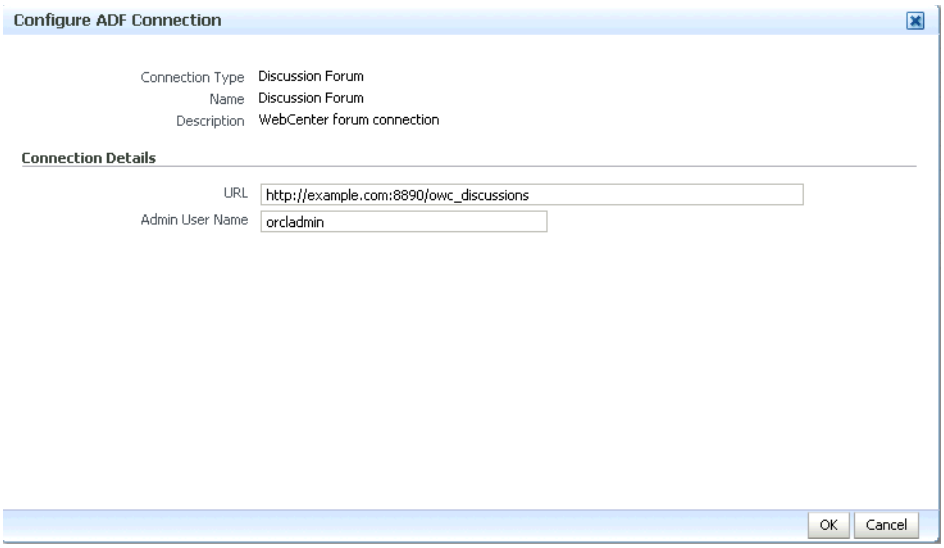
Figure 42–12 Configure ADF Connections Page



12. Click the **edit** icon for each connection and check that the connection settings are correct for the target environment (for example, staging or production).

    For a Discussion Forum connection (shown in Figure 42–13), for example, ensure that the URL to the Discussions server, and the user account used to connect to the server are correct for the target environment.

Figure 42–13 Discussion Forum Connection Settings



For WSRP producers, two connections are shown for each producer: a WSRP Producer and a Web Service connection. Typically only the Web Service connection must be changed to the target producer, and this contains four URL endpoints, all of which must be changed. The WSRP Producer connection only configures proxy settings that can be set independent of the default proxy setting for the application server, if this is required.

If any connections to portlet producers in the EAR file must be changed to point to producers in the target deployment environment, it is important to change them here. This ensures the portlet customizations are imported to the target producers as the application starts. For more information, see Section 42.1.6, "Deploying the Application to a WebLogic Managed Server".

Note: If any target producers are not reachable as the application starts for the first time, the import fails. After the portlet producer becomes reachable, restart the application and try to import again.

If you do not modify producer connections using the Configure ADF Connections page and they are pointing to incorrect but reachable producer locations (for example, a producer in a development environment), portlets are imported to the incorrect producers.

To correct this, after deployment use Fusion Middleware Control (see Section 21.2.1, "Registering a WSRP Producer Using Fusion Middleware Control" and Section 21.4.1, "Registering an Oracle PDK-Java Producer Using Fusion Middleware Control") or WLST commands (see Section 21.2.2, "Registering a WSRP Producer Using WLST" or Section 21.4.2, "Registering an Oracle PDK-Java Producer Using WLST") to modify the producer URL endpoint, and then redeploy the application as described in Section 42.3.2, "Redeploying Portal Framework Applications Using Fusion Middleware Control."

---

13. If required, specify additional deployment options such as the Web modules to include in your application or security migration settings.

14. In the Deployment Plan section, click **Edit Deployment Plan** to optionally edit the currently selected Deployment Plan.
15. In the Deployment Plan section, click **Save Deployment Plan** to optionally save the currently selected Deployment Plan for reuse when you redeploy the application.
16. To start the deployment process, click **Deploy**.

    Fusion Middleware Control displays processing messages.
17. Click **Close** in the Deployment Succeeded page.

    The Portal Framework application (and its deployment plan) is now deployed on the WebLogic Managed Server instance.
18. If you restart the WebLogic Managed Server on which you deployed the application during your Fusion Middleware Control session, refresh the Farm from the Farm menu to update the application status.

> **Note:** If you configured connections during deployment, these are not stored as part of the deployment plan. You must specify these connection details again the next time you deploy.

42.1.6.5 Deploying Applications Using WLST

To deploy a Portal Framework application using the WLST command line, WLST must be connected to the Administration Server. You must invoke the `deploy` command on the computer that hosts the administration server.

To deploy a Portal Framework application using WLST:

1. Start the WLST shell.

   For information on starting the WLST shell, see Section 1.13.3, "Oracle WebLogic Scripting Tool (WLST)."
2. Connect to the Administration Server of your Oracle WebCenter Portal installation:

   ```
   connect("user_name","password","host_name:port")
   ```

   Where:

   - *user_name* is the user name to access the Administration server (for example, `weblogic`).
   - *password* is the password to access the Administration server (for example, `weblogic`).
   - *host_name* is the host name of the Administration Server (for example, `myserver.example.com`).
   - *port* is the port number of the Administration Server (`7001` by default)

     You should see the following message:

     ```
     Successfully connected to Admin Server 'AdminServer' that belongs to domain
     'wc_domain'.
     ```
3. Retrieve the MDS configuration by running the following command:

   ```
   archive = getMDSArchiveConfig(fromLocation='ear_file_path')
   ```

Where *ear_file_path* is the path and file name of the EAR file you are deploying (for example, `/tmp/myEarFile.ear`). For more information, see the `getMDSArchiveConfig` command in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

4. After retrieving the MDS configuration information from the EAR file, you must set the proper MDS schema information according to your Oracle WebCenter Portal setup (for example, your application might be using a database connection based on a specific schema). To set the MDS schema information, run the following command:

   ```
   archive.setAppMetadataRepository(repository='respository',partition='partition'
   ,type='DB',jndi='jndi')
   ```

   Where:

   - *repository* is the name of the database schema (for example, `mds-Feb23demo`)
   - *partition* is the individual entity in the repository to allow each application to have its own namespace (for example, `webcenter`).
   - *jndi* is the path and name used to allow access by the application server's other components (for example, `jdbc/mds/Feb23demo`)

5. After setting the MDS repository information, save the MDS configuration information with the following command:

   ```
   archive.save()
   ```

6. Deploy the Portal Framework application using the WLST deploy command.

   ```
   deploy(app_name, path, [targets] [stageMode], [planPath], [options])
   ```

   Where:

- *appName* is the name of the Portal Framework application to be deployed (for example, `composerWLSTApp`).
- *path* is the path to the EAR file to be deployed (for example, `/tmp/customApp.ear`).
- *targets* specifies the target Managed Server(s) to which to deploy the application (for example, `CustomAppServer`). You can optionally list multiple comma-separated targets. To enable you to deploy different modules of the application archive on different servers, each target may be qualified with a module name, for example, `module1@server1`. This argument defaults to the server to which WLST is currently connected.
- *[stageMode]* optionally defines the staging mode for the application you are deploying. Valid values are `stage`, `nostage`, and `external_stage`.
- *[planPath]* optionally defines the name of the deployment plan file. The file name can be absolute or relative to the application directory. This argument defaults to the `plan/plan.xml` file in the application directory, if one exists.
- *[options]* is an optional comma-separated list of deployment options, specified as name-value pairs. For more information about valid options, see the WLST deploy command in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

When you see the following message, the application has been successfully deployed and is ready to be accessed:

```
Completed the deployment of Application with status completed
```

> **Note:** Since WLST does not prompt you to modify connections during deployment, the connection information in the EAR file is used to identify the target producer location in the last start-up. If that location is unreachable, correct the location after deploying the application by bringing up the target producers and restarting the application. Migration of portlet customizations starts automatically.
>
> If the producer connections point to incorrect producers (for example, development producers), and those producers are reachable, the migration of portlet customizations starts using those producers. Since the migration completes, although incorrectly, restarting the application does not automatically restart the migration process.
>
> To remedy this, after deployment, use Fusion Middleware Control (see Section 21.2.1, "Registering a WSRP Producer Using Fusion Middleware Control" and Section 21.4.1, "Registering an Oracle PDK-Java Producer Using Fusion Middleware Control") or WLST commands (see Section 21.2.2, "Registering a WSRP Producer Using WLST" or Section 21.4.2, "Registering an Oracle PDK-Java Producer Using WLST") to modify the producer URL endpoint, and then redeploy the application as described in Section 42.3.2, "Redeploying Portal Framework Applications Using Fusion Middleware Control."

42.1.6.6 Deploying Applications Using the WLS Administration Console

You can use the WLS Administration Console to deploy a Portal Framework application or a portlet producer application. However, the Console does not offer a means to change ADF connections, including the essential MDS connection. To use the Console to deploy a Portal Framework application, the MDS connection in the EAR file must be configured to the target deployment repository. Follow steps 1-5 in Section 42.1.6.5, "Deploying Applications Using WLST," then follow the steps below to deploy a Portal Framework application or portlet producer application using the WLS Administration Console.

> **Note:** Oracle does not recommend deploying Portal Framework applications to any of the preconfigured Managed Servers created during the installation, or to the Administration Server. For Portal Framework applications created in JDeveloper, follow the process described in the "Extending an Existing Domain" section in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal* to create and provision a new Managed Server before deploying. For portlet producer applications, you can create a Managed Server instance, or optionally deploy to the `WC_Portlet` server.

To deploy a Portal Framework application or portlet producer application using the WLS Administration Console:

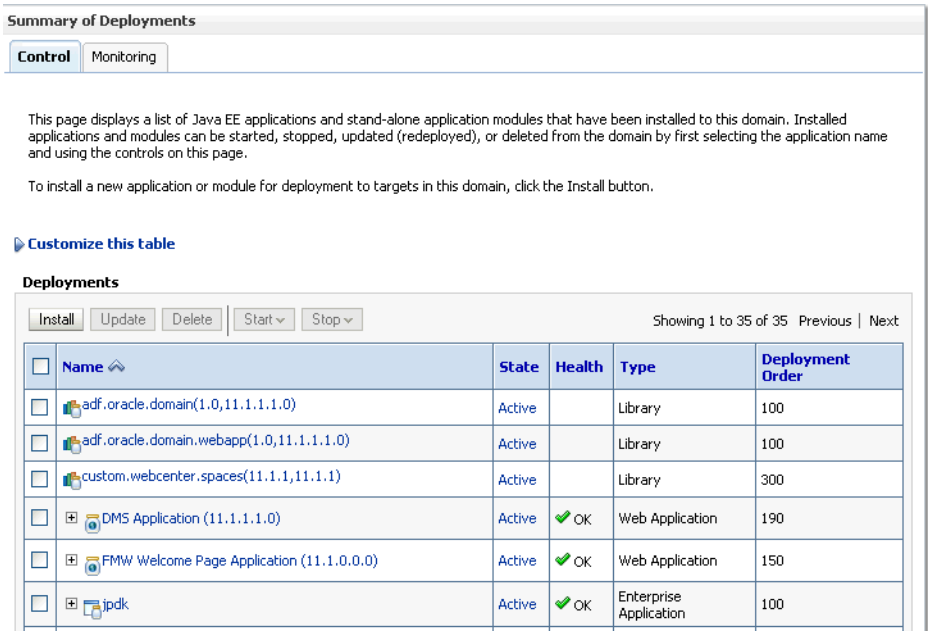
1. Log in to the WLS Administration Console.

   For information on logging into the WLS Administration Console, see Section 1.13.2, "Oracle WebLogic Server Administration Console."

2. In the Domain Structure pane, click **Deployments**.

   The Summary of Deployments page displays (see Figure 42–14).

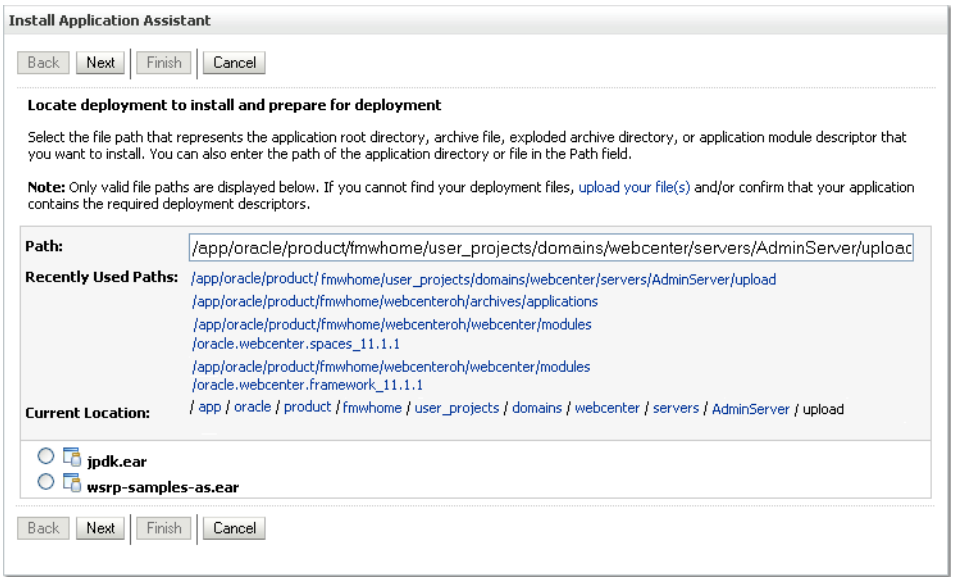
Figure 42–14 Deployment Summary



3. On the Deployment Summary pane, click **Install**.

   The Install Application Assistant page displays (see Figure 42–15).

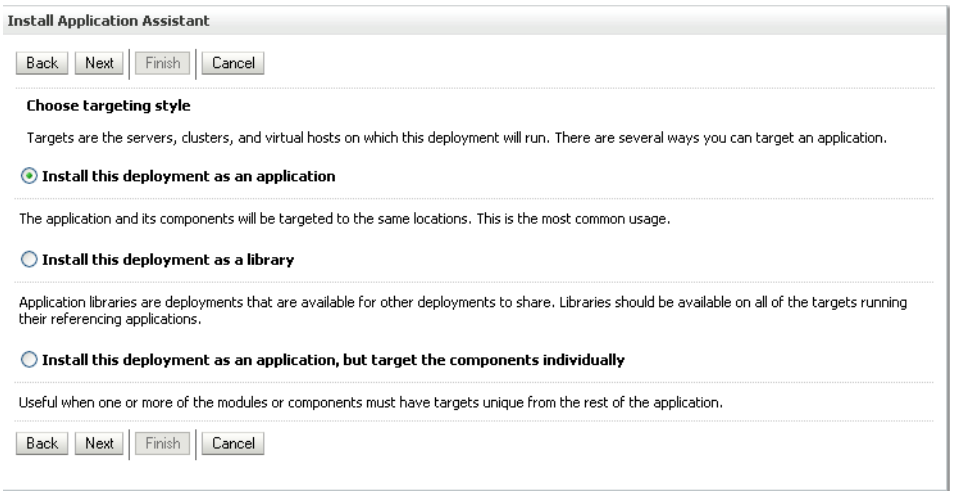
Figure 42–15 Install Application Assistant Page



4. Using the Install Application Assistant **Path** field, locate the EAR file that corresponds to the Web application or portlet producer application you want to install. Select the EAR file and click **Next**.

   Page 2 of the Install Application Assistant page displays (see Figure 42–16).

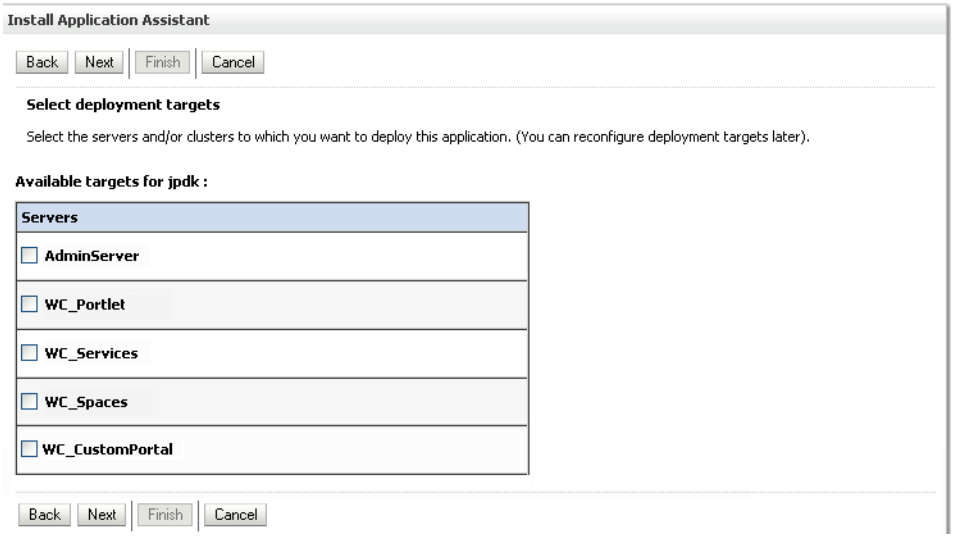
Figure 42–16 Install Application Assistant - Page 2



5. Select **Install this deployment as an application** (for both Portal Framework applications and portlet producers) and click **Next**.

   Page 3 of the Install Application Assistant displays (see Figure 42–17).

Figure 42–17 Install Application Assistant - Page 3



6. Select the deployment target to which to deploy the Web application and click **Next**.

7. Review the configuration settings you specified, and click **Finish** to complete the installation.

   To change a producer URL after deployment, use Fusion Middleware Control (see Section 21.2.1, "Registering a WSRP Producer Using Fusion Middleware Control" and Section 21.4.1, "Registering an Oracle PDK-Java Producer Using Fusion Middleware Control") or WLST commands (see Section 21.2.2, "Registering a WSRP Producer Using WLST" or Section 21.4.2, "Registering an Oracle PDK-Java Producer Using WLST") to modify the producer URL endpoint, and then redeploy the application as described in Section 42.3.2, "Redeploying Portal Framework Applications Using Fusion Middleware Control."

42.1.6.7 Saving and Reusing the Deployment Plan

A deployment plan contains the configuration data needed to deploy an archive to a Managed Server. You can create a deployment plan while you're building and testing your application, or when you deploy your EAR file using Fusion Middleware Control as described in Section 42.1.6.4, "Deploying Applications Using Fusion Middleware Control." If there are deployment descriptors packaged within the EAR file, the deployment uses the data in these files. If you need to make any changes to the `web.xml` file, Oracle recommends that you create a deployment plan.

Once created, a deployment plan can be saved as part of the application properties on the target Managed Server, and re-used when redeploying the application using Fusion Middleware Control, as described in Section 42.3.2, "Redeploying Portal Framework Applications Using Fusion Middleware Control," or using WLST as described in Section 42.3.3, "Redeploying Portal Framework Applications Using WLST."

42.1.7 Migrating Customizations and Data Between Environments

You can export and import customizations made to pages, tools and services, and portlets (PDK-Java and WSRP version 2 producers) of a deployed application. For more information, see Chapter 44.1, "Exporting and Importing Portal Framework Applications for Data Migration."

42.1.8 Configuring Applications to Run in a Distributed Environment

For information about configuring your Portal Framework application to run in a distributed environment, see *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle WebCenter Portal*, and the "Configuring High Availability for Oracle ADF and Oracle WebCenter Portal" chapter in *Oracle Fusion Middleware High Availability Guide*.

42.2 Undeploying Portal Framework Applications

This section describes how to undeploy a Portal Framework application or portlet producer application using Fusion Middleware Control, or from the command line using WLST.

> **Note:** When a Portal Framework application is undeployed, its application credentials and MDS customizations are kept in case the application is redeployed to the same domain. If the application will not be redeployed in this domain, or if it is important to reset these back to initial conditions before the next deployment, then after undeploying an application you can remove the application's credential map from the Credential Store as described in Section 42.2.3, "Removing an Application's Credential Map." You can also remove the MDS repository partition as described in the "Deleting a Metadata Partition from a Repository" section in *Oracle Fusion Middleware Administrator's Guide*.

This section contains the following subsections:

- Section 42.2.1, "Undeploying Portal Framework Applications Using Fusion Middleware Control"
- Section 42.2.2, "Undeploying Portal Framework Applications Using WLST"

- Section 42.2.3, "Removing an Application's Credential Map"

42.2.1 Undeploying Portal Framework Applications Using Fusion Middleware Control

This section describes how to undeploy a Portal Framework application using Fusion Middleware Control.

To undeploy a Portal Framework application using Fusion Middleware Control:

1. Log in to Fusion Middleware Control.

   See Section 6.1, "Displaying Fusion Middleware Control Console."

2. From the Navigation pane, expand **Application Deployments**, then click the application that you want to undeploy.
3. From the Application Deployment menu, select **Application Deployment > Undeploy**.
4. On the confirmation page, click **Undeploy**.
5. When the operation completes, click **Close**.

42.2.2 Undeploying Portal Framework Applications Using WLST

This section describes how to undeploy a Portal Framework application using WLST.

To undeploy a Portal Framework application using WLST:

1. Start the WLST shell.

   For information on starting the WLST shell, see Section 1.13.3, "Oracle WebLogic Scripting Tool (WLST)."

2. Connect to the Administration Server of your WebCenter Portal installation:

   ```
   connect("user_name","password","host_name:port")
   ```

   Where:

   - `user_name` is the user name to access the Administration Server (for example, `weblogic`).
   - `password` is the password to access the Administration Server (for example, `weblogic`).
   - `host_name` is the host name of the Administration Server (for example, `myserver.example.com`).
   - *`port`* is the port number of the Administration Server (`7001` by default).

     You should see the following message:

     ```
     Successfully connected to Admin Server 'AdminServer' that belongs to domain
     'wc_domain'.
     ```

3. Use the `undeploy` command to undeploy the application:

   ```
   undeploy(app_name,[targets],[options])
   ```

   Where:

   - `app_name` is the deployment name for the deployed application.
   - `[targets]` is a list of the target servers from which the application will be removed. Optional. If not specified, defaults to all current targets.

- `[options]` is a comma-separated list of deployment options, specified as name-value pairs. Optional. See the `deploy` command for a complete list of options.

42.2.3 Removing an Application's Credential Map

When a Portal Framework application is undeployed, its application credentials are not removed. Consequently, you must manually remove the credential map used for the application after it is undeployed using Fusion Middleware Control.

To remove an application's credentials map using Fusion Middleware Control:

1. Determine the credentials map name used by the application by inspecting the contents of the application's `adf-config.xml` and locating the value for `adfAppUID`. For example:

   ```
   <adf:adf-properties-child
   xmlns="http://xmlns.oracle.com/adf/config/properties">
   <adf-property name="adfAppUID" value="Veeva-7209"/>
   </adf:adf-properties-child>
   ```

   In this case, **`Veeva-7209`** is the credential map name used by the application.

2. Log in to Fusion Middleware Control.

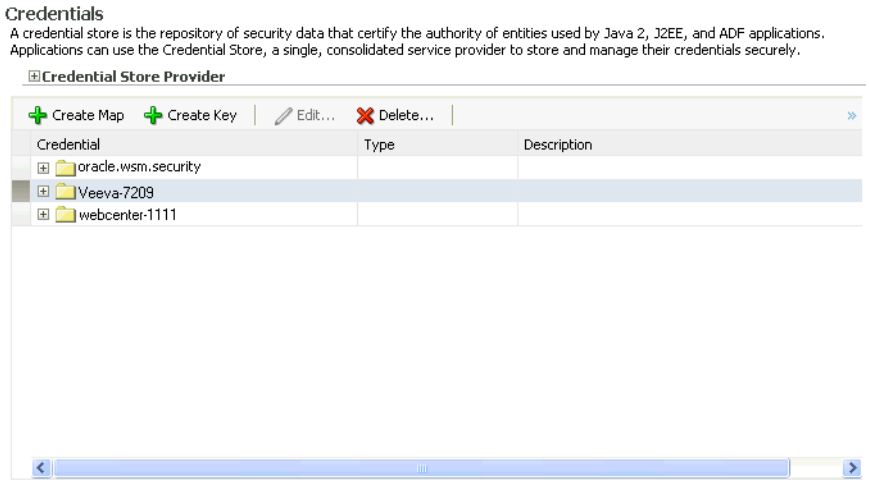
   For information on logging into Fusion Middleware Control, see Section 6, "Starting Enterprise Manager Fusion Middleware Control."

3. In the Navigation pane, expand the **WebLogic Domain** node and select the target domain (for example, `wc_domain`).

4. From the WebLogic Domain drop-down menu, select **Security > Credentials**.

   The Credentials pane displays (see Figure 42–18).

Figure 42–18 Credentials Pane



5. Select the credential map to remove and click **Delete**.

6. Click **Yes** to confirm deleting the credential map.

42.3 Redeploying Portal Framework Applications

This section describes how to redeploy a Portal Framework application using Fusion Middleware Control or from the command line using WLST. When you redeploy a new version of an application, you cannot change:

- the application's deployment targets
- the application's security model

To change deployment targets or application security settings, you must first undeploy the active version of the application. For information on how to undeploy an application, see Section 42.2, "Undeploying Portal Framework Applications".

> **Note:** Some system .EAR files, such as `wcps-services.ear` and `wsrp-tools-as.ear`, are not versioned and were not intended to be redeployed. Redeploying these files will generate an error.

This section contains the following subsections:

- Section 42.3.1, "Redeployment Considerations"
- Section 42.3.2, "Redeploying Portal Framework Applications Using Fusion Middleware Control"
- Section 42.3.3, "Redeploying Portal Framework Applications Using WLST"

42.3.1 Redeployment Considerations

In most cases, when redeploying an application, you want to preserve any changes to application data. Three important pieces of information about an application can be altered after deployment during run-time:

- Application Configuration -- which includes connection information.
- Application Metadata -- which includes the customizations and personalizations on the application itself, such as those created when user edits a page and adds content to it.
- Portlets Preferences-- which includes customizations and personalizations of the portlet instances.

> **Note:** You must delete runtime customizations (customizations not done through JDeveloper) before redeploying an updated application that has had major changes to artifacts such as pages, connections, or task flows.

The following subsections explain how to preserve these three types of information about an application:

- Section 42.3.1.1, "Preserving Application Configuration"
- Section 42.3.1.2, "Preserving Service and User Customizations"
- Section 42.3.1.3, "Preserving Asset Customizations"
- Section 42.3.1.4, "Preserving Portlet Customizations"

Note: To preserve application information, you must redeploy using the same MDS partition that was used or created using the initial deployment.

42.3.1.1 Preserving Application Configuration

In most cases, the end-points of tools and services and portlet-producers are different in a test or staging environment than in a production environment. Therefore, when an application is redeployed to a production environment, you must reconfigure the application to work with the production environment services and producers or reuse the configuration used previously. Fusion Middleware facilitates this by storing the configuration information in the MDS repository.

When you deploy the application for the first time, the base document of the application configuration is created in the MDS repository. This configuration is the set of all of the application's connections and their properties that are packaged in the EAR file. After the deployment, you may need to modify the connections using Fusion Middleware Control or WLST in response to production needs. This reconfiguration creates a layer of customization for the configuration changes in the MDS repository.

When you redeploy the application, the configuration packaged with the application is laid down as the base document, but the customizations to the configuration are preserved. Therefore, the application's redeployment settings match the most recent configuration performed.

However, customizations are completely preserved only when there are no changes in the base document. If you redeploy an application where the packaged connection information has changed, the following can be expected:

- A new connection is added to the packaged configuration.
  The new connection should display without problems.
- A connection has been removed in the packaged configuration.
  If you configured this connection after the last deployment, then the connection does not display after deployment, and you must re-create it.
- A connection property has been changed in the packaged configuration.
  The customized properties are used. Connection customizations are managed at the individual connection level, and not at the properties level.

42.3.1.1.1 Preserving Configuration Across Deployment Using WLST

If you use WLST commands to configure Portal Framework application on a stage instance, you can easily combine them into a script and then use a variant of that script to re-create the connections on a production instance. Using this approach, you can always reconfigure an application to the target configuration without worrying about the details in the packaged configuration.

42.3.1.2 Preserving Service and User Customizations

Application metadata can change post-deployment due to customizations done by users at runtime. When you redeploy the application, in most circumstances, you must preserve this customization information so that users see exactly what they were seeing before.

Application and user customizations are stored in the MDS repository, and the same rules apply for preserving application metadata as for preserving configuration settings.

When the application is redeployed, the base documents for all application artifacts are replaced with what is packaged in the EAR file. However, customizations are retained. There is no impact to this information unless the base artifact is changed, in which case the same rules apply as for configuration settings, which are:

- If new elements are added to the package, then they appear as they are.
- If elements are removed from the package, for which customizations were created, those customizations are ignored.
- If elements are changed, then the effect depends on what exactly is changed, but must be verified.

> **Best Practice Note:** In some cases, you may want to export all application and user customizations in a production application instance and import it into a test or staging instance. You can then test the application against those customizations to see that the new changes do not have an undesired impact.

42.3.1.3 Preserving Asset Customizations

Users can create new assets at runtime using the Assets page. If you plan to redeploy the application and want to preserve runtime-created assets, before redeployment you must first download the assets from the running application and import the resulting archive file into the design-time environment.

If you do not download and import runtime-created assets, they are lost upon redeployment of the application. Any new pages created at runtime that use the lost assets are still available even though the assets themselves are no longer available in the Assets page. This is because the `generic-site-resources.xml` file, which is updated at runtime when new assets are created, is overwritten on redeployment by the design-time version of the file.

42.3.1.4 Preserving Portlet Customizations

Portlet customizations are packaged with the metadata in the EAR file. Application startup after deployment kicks off the portlet customization migration to the target producers. The target producers are identified by resolving connection customizations. If you have modified your producer connections before redeployment, then those modified connections are used to identify target producers. Note that if you redeploy an EAR file with the same checksum (that is, the same file) as the pre-existing one, portlet customization and personalizations are not overwritten.

42.3.2 Redeploying Portal Framework Applications Using Fusion Middleware Control

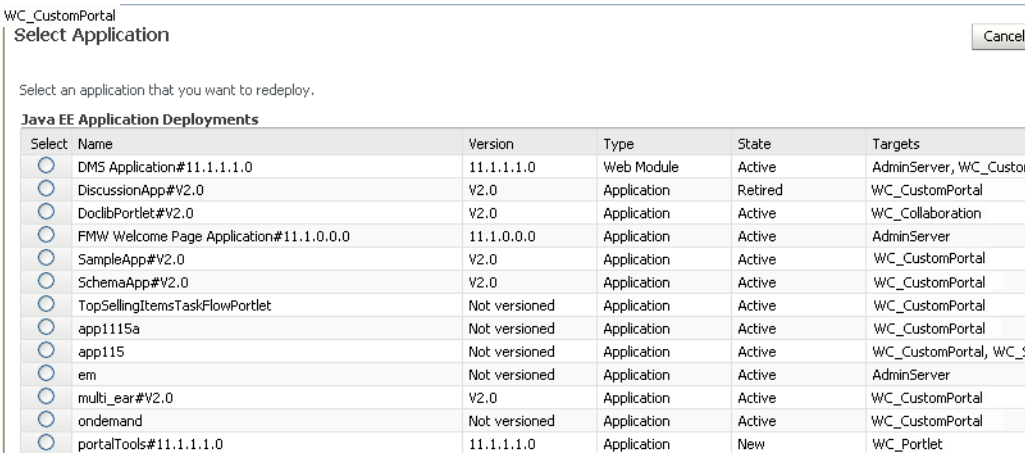
This section describes how to redeploy a Portal Framework application using Fusion Middleware Control.

To redeploy a Portal Framework application using Fusion Middleware Control:

1. Log in to Fusion Middleware Control. For more information, see Section 6.1, "Displaying Fusion Middleware Control Console."
2. From the Navigation pane, expand the farm, then **WebLogic Domain**, and then the domain.
3. Select the server to which to redeploy the application, and then right-click and select **Application Deployment - Redeploy** from the menu.

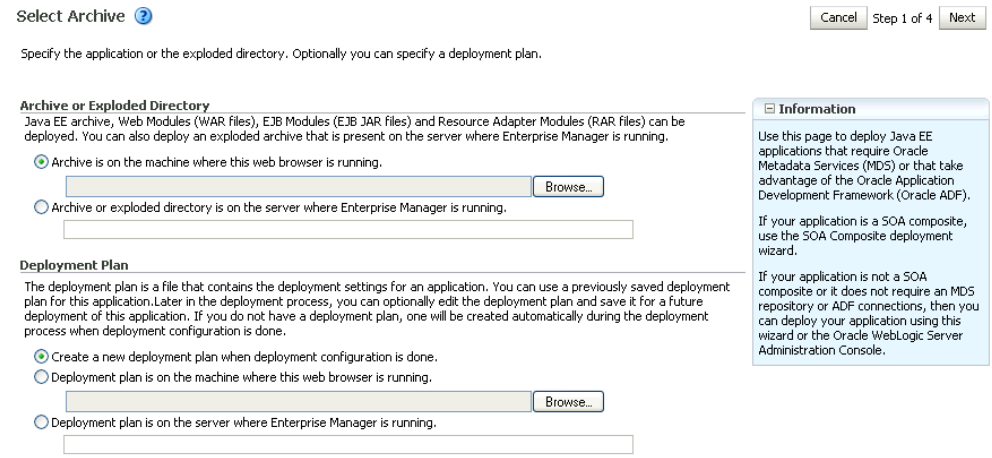
   The Select Application page displays (see Figure 42–19).

Figure 42–19 Select Application Page



4. Select the application that you want to redeploy.
5. Click **Next** to display the Select Archive page (see Figure 42–20).

Figure 42–20 Select Archive Page

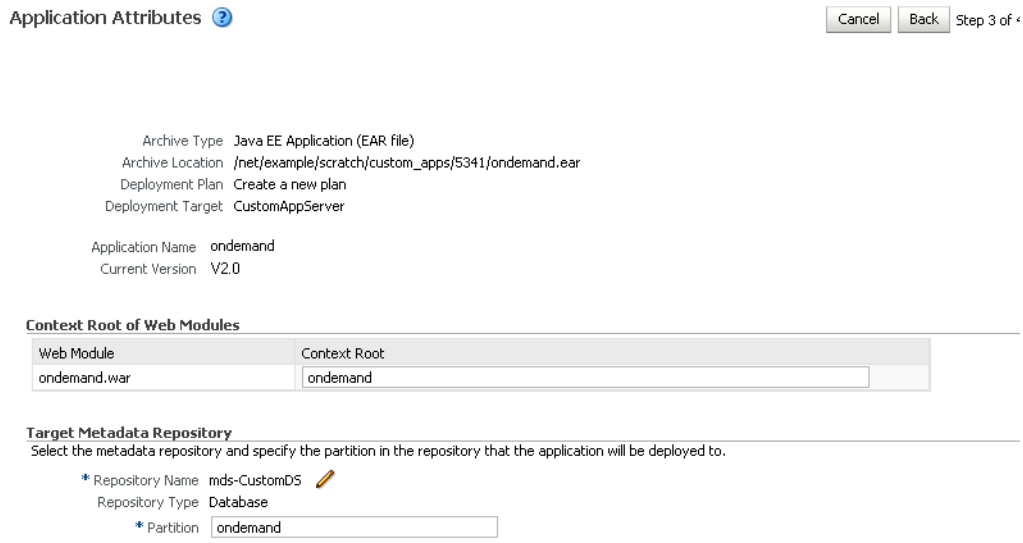


6. In the Archive or Exploded Directory section, do one of the following:
   - Select **Archive is on the machine where this web browser is running** and enter the location of the archive or click **Browse** to find the archive file.
   - Select **Archive or exploded directory is on the server where Enterprise Manager is running** and enter the location of the archive or click **Browse** to find the archive file.
7. In the Deployment Plan section, do one of the following:
   - Select **Create a new deployment plan when deployment configuration is done** to automatically create a deployment plan after the redeployment process.
   - Select **Deployment plan is on the machine where this web browser is running** and enter the path to the plan or click **Browse** to find the plan.
   - Select **Deployment plan is on the server where Enterprise Manager is running** and enter the path to the plan or click **Browse** to find the plan.

8. Click **Next**.

   The Application Attributes page displays (see Figure 42–21).

Figure 42–21 Application Attributes Page



9. In the Context Root of Web Modules section, specify the context root for your application if you have not specified it in `application.xml`. The context root is the URI for the web module. Each web module or EJB module that contains web services may have a context root.

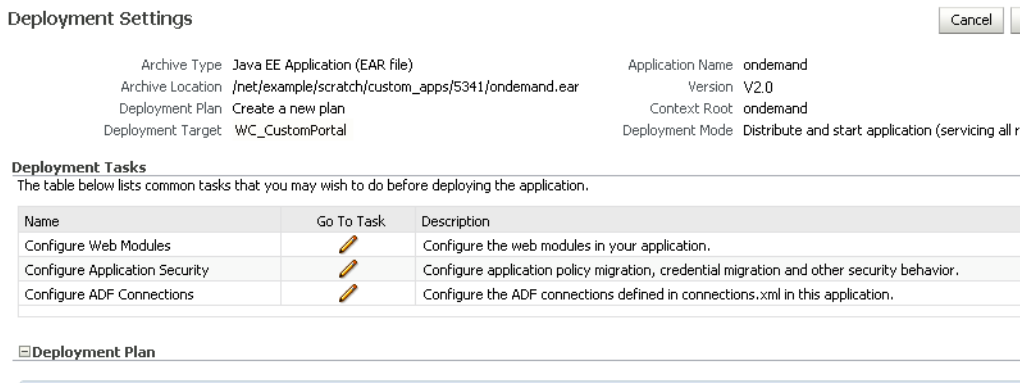
10. In the Target Metadata Repository section, select the MDS repository and enter the **Partition**.

> **Caution:** Be careful to use the same repository connection and partition name that you used when you originally deployed the application. If you do not, all customizations are lost.

11. Click **Next**.

    The Deployment Settings page displays (see Figure 42–22).

Figure 42–22 Deployment Settings Page

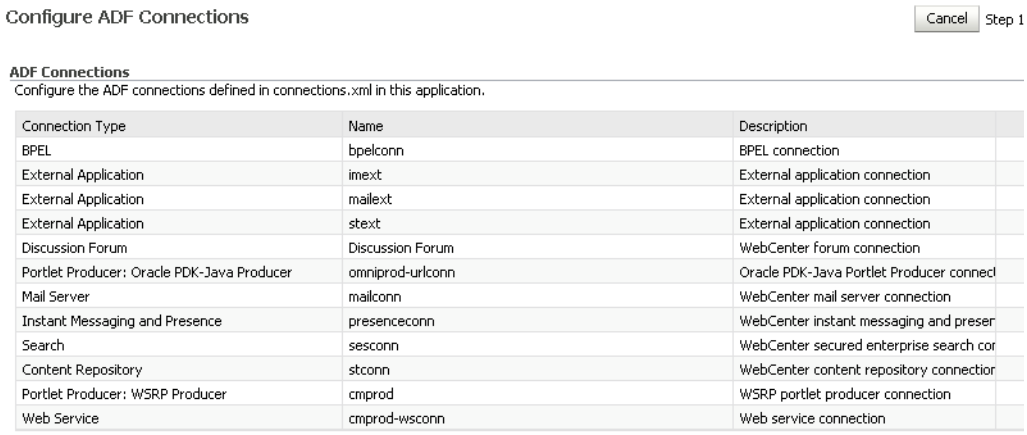


12. On this page, you can perform common tasks before deploying your application, such as configuring connections, or you can edit the deployment plan or save it to a disk. You can:
    - Configure web modules
    - Configure application security for application roles and policies
    - Configure ADF connection settings
13. Click the **edit** icon for Configure ADF Connections to check connection settings associated with the Portal Framework application.

    Note: Editing ADF Connections is only necessary for connections not set after a prior deployment. Any connections configured after a prior deployment will override settings you make during this step.

    The Configure ADF Connections page displays (see Figure 42–23).

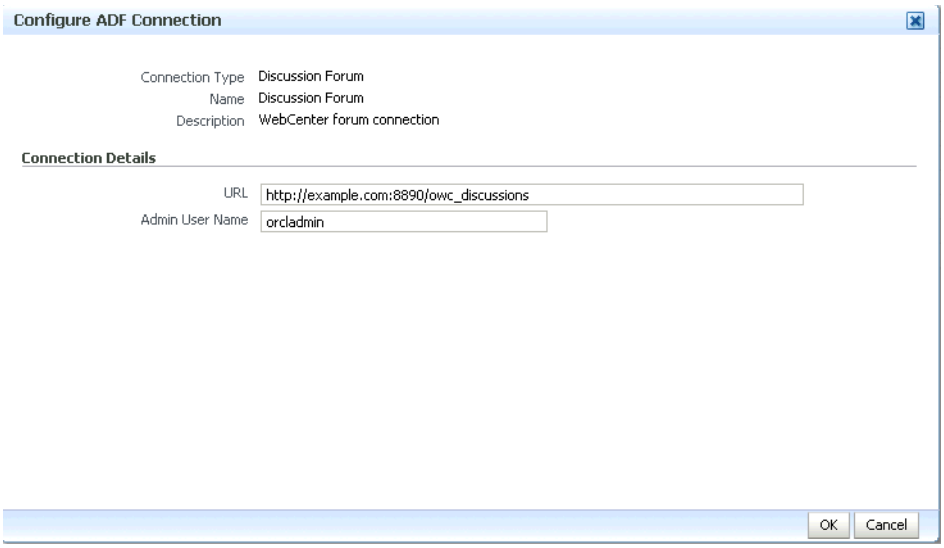
Figure 42–23 Configure ADF Connections Page



14. Click the **edit** icon for each connection and check that the connection settings are correct for the target environment (for example, staging or production).

    For a Discussion Forum connection (shown in Figure 42–24), for example, ensure that the URL to the discussions server, and the user account used to connect to the server are correct for the target environment.

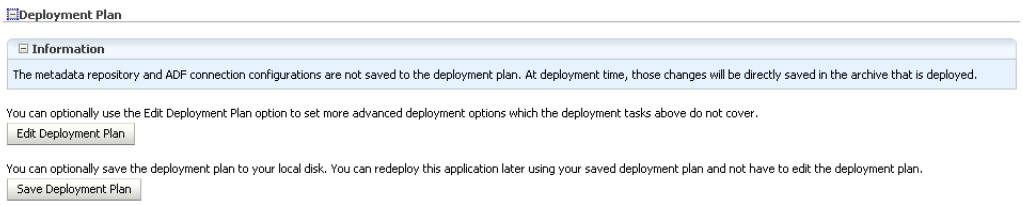
Figure 42–24 Discussion Forum Connection Settings



15. If required, specify additional deployment options such as the Web modules to include in your application or security migration settings.
16. Expand Deployment Plan.

    The Deployment Plan settings display (see Figure 42–25).

Figure 42–25 Deployment Settings Page - Deployment Plan Section



    You can edit and save the deployment plan to your local hard drive, if you choose, so that you can use those settings to redeploy the application again later. See Section 42.1.6.7, "Saving and Reusing the Deployment Plan" for more information about deployment plans.
17. Click **Redeploy**.
18. When the redeployment completes, click **Close**.

> **Note:** If you restart the WebLogic Managed Server on which you deployed the application during your Fusion Middleware Control session, refresh the Farm from the Farm menu to update the application status.

42.3.3 Redeploying Portal Framework Applications Using WLST

To redeploy a Portal Framework application using the WLST command line, WLST must be connected to the administration server. You must invoke the `redeploy` command on the computer that hosts the administration server.

To redeploy a Portal Framework application using WLST:

1. Start the WLST shell.

   For information on starting the WLST shell, see Section 1.13.3, "Oracle WebLogic Scripting Tool (WLST)."

2. Connect to the administration server of your Oracle WebCenter Portal installation:

   ```
   connect("user_name","password","host_name:port")
   ```

   Where:

   - *user_name* is the user name to access the administration server (for example, weblogic).
   - *password* is the password to access the administration server (for example, weblogic).
   - *host_name* is the host name of the administration server (for example, myserver.example.com).
   - *port* is the port number of the Administration Server (7001 by default).

     You should see the following message:

     ```
     Successfully connected to Admin Server 'AdminServer' that belongs to domain
     'wc_domain'.
     ```

3. Use the redeploy command to redeploy the application:

   ```
   redeploy(app_name,[planPath],[options])
   ```

   Where:

   - *app_name* is the deployment name for the application to redeploy.
   - *[planPath]* Name of the deployment plan file. The filename can be absolute or relative to the application directory. Optional. This argument defaults to the plan/plan.xml file in the application directory, if one exists.
   - *[options]* is a comma-separated list of deployment options, specified as name-value pairs. Optional. See the deploy command for a complete list of options.

42.4 Post-Deployment Configuration

After your Portal Framework application is deployed, you must check that the settings that were deployed are valid for the target Managed Server. Settings to check include those for security, connections, and data sources.

This section includes the following subsections:

- Section 42.4.1, "Checking Security Configurations After Deployment"
- Section 42.4.2, "Checking Application Connections After Deployment"
- Section 42.4.3, "Checking Data Source Connections"
- Section 42.4.4, "Tuning the Application"

42.4.1 Checking Security Configurations After Deployment

Before deploying your application you must set up the Identity Store and the Policy and Credential Store on the target Managed Server. After deployment, check that the

application configurations match those of the target server. You should also check that all other applicable post-deployment security configurations, such as SSL and single sign-on, have been properly configured, as described in Section 30.2.5, "Post-deployment Security Configuration Tasks."

42.4.2 Checking Application Connections After Deployment

After deploying your Portal Framework application, check that all of the connections used by your application have been properly set. Connections that you may have to configure or reconfigure include connections for:

- BPEL worklists
- External applications
- Discussions server
- Mail server
- Instant Messaging and Presence (IMP) server
- Search
- WSRP producers
- PDK-Java portlet producers
- Web Services
- Content repositories
- Personal event server
- Analytics collector

42.4.3 Checking Data Source Connections

After deploying your Portal Framework application to a custom Managed Server, check that the data sources that you configured during testing are still valid for the deployed application. For information on how to configure data sources for the Metadata Services (MDS) repository your Portal Framework application, see the "Configuring JDBC Data Sources" chapter in *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server*. Note that when setting up the data source, you must provide a password or the connection may not be created when the application is deployed.

42.4.4 Tuning the Application

After your Portal Framework application has been deployed and correctly configured, check the system file limit, data source settings, and JRockit virtual machine (JVM) arguments as described in Section 27.5, "Tuning Oracle WebCenter Portal Performance". Also see the "Oracle WebCenter Portal Performance Tuning" chapter in *Oracle Fusion Middleware Performance and Tuning Guide*, and Section 27, "Monitoring Oracle WebCenter Portal Performance" for information on how to diagnose performance problems.

43

Administering Portal Framework Applications Using the Administration Console

This chapter provides information about the runtime administration console that is available for Portal Framework applications. Portal Framework applications are portal applications built using the WebCenter Portal Framework application template in Oracle JDeveloper

This chapter includes the following topics:

Permissions: To perform the tasks in this chapter, you must be granted the `Administrator` role through the Portal Framework application's Administration Console. For details, see Section 43.4.2, "Understanding Application Roles and Permissions"

See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

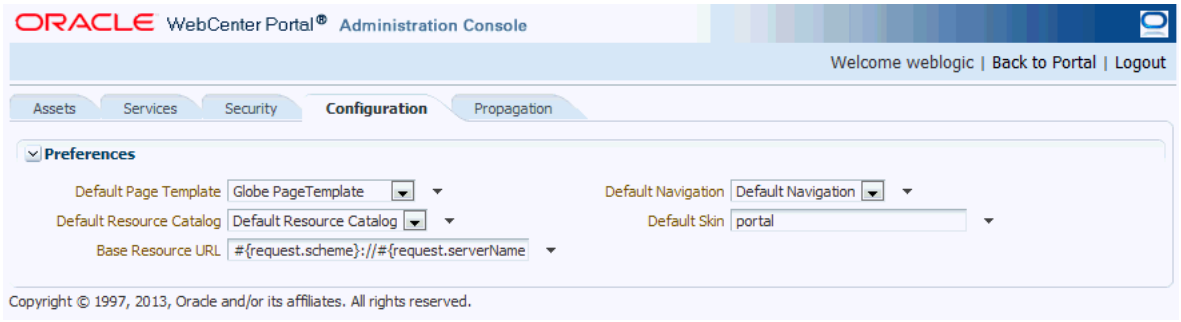
43.1 Introduction to the Administration Console for Portal Framework Applications

By default, Portal Framework applications offer several administration pages (Figure 43–1) that enable authenticated administrators to perform common administrative duties, including:

- Setting application-level preferences

- Managing users and granting application roles
- Managing and configuring application assets
- Managing content
- Managing and configuring portlet producers
- Managing and configuring external applications
- Creating and managing polls
- Propagating application updates

Figure 43–1 WebCenter Portal Administration Console



43.2 Accessing the Administration Console for Portal Framework Applications

To access the administration console for a Portal Framework application:

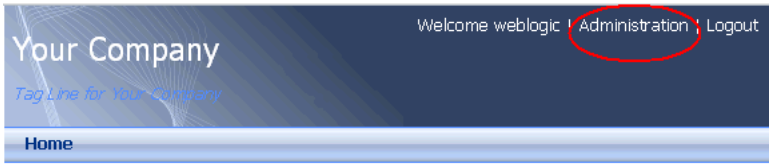
1. Log in to your application as an administrator.

   Initially, the WebCenter Portal Administration Console is only available to the system administrator. For information on how to grant the `Administrator` role to others, see Section 43.4.3.3, "Giving a User Administrative Privileges."

2. Do one of the following:

   - Click the **Administration** link (Figure 43–2).

Figure 43–2 WebCenter Portal Administration Console - Administration Link



   - Access the WebCenter Portal Administration Console using the direct URL:

     The default direct URL is:

     `http://www.server:port/context_root/admin`

     For example: `http://mycompany.com:8888/myapp/admin`

     Application developers can, however, customize the direct URL using the `web.xml` entry:

     ```
     <servlet-mapping>
     ```

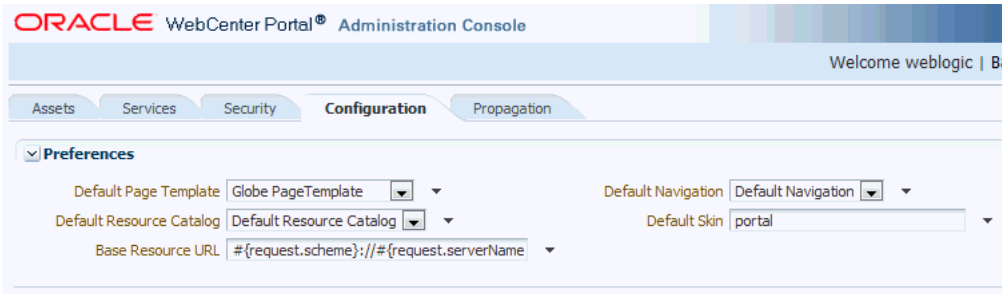
```
  <servlet-name>PortalAdminServlet</servlet-name>
  <url-pattern>/admin</url-pattern>
</servlet-mapping>
```

The WebCenter Portal Administration Console displays (see Figure 43–1).

43.3 Configuring Defaults for Portal Framework Applications

On deployment, Portal Framework applications are pre-configured with various default settings which administrators can customize to suit their audience through the WebCenter Portal Administration Console. From the **Configuration** tab (Figure 43–3), you can specify a default page template, skin, resource catalog, and navigation model.

Figure 43–3 WebCenter Portal Administration Console - Configuration Tab



This section includes the following topics:

- Section 43.3.1, "Choosing a Default Page Template"
- Section 43.3.2, "Choosing Default Resource Catalogs"
- Section 43.3.3, "Choosing a Default Navigation"
- Section 43.3.4, "Choosing a Default Skin"
- Section 43.3.5, "Choosing the Default Base Resource URL"

43.3.1 Choosing a Default Page Template

In a Portal Framework application, page templates define how individual pages and groups of pages display on a user's screen. Every page displays within a page template. For more information, see the "Developing Page Templates" chapter in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

Administrators can define the default page template that is used to display pages.

To select a default page template for the application:

1. Navigate to the **Configuration** administration tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. Choose a **Default Page Template** from the list provided.

   The template you select is applied to existing pages and any new pages that are created.

Note: If users create pages and hard-code links to page templates, the **Default Page Template** has no effect.

---

3. Click **Apply**.

43.3.2 Choosing Default Resource Catalogs

In a Portal Framework application, the resource catalog specifies a collection of elements, such as layout components, task flows, portlets, documents, and others, that authorized users can add to the application at runtime. For more information, see the "Developing Resource Catalogs" chapter in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

To select a default resource catalog for the application:

1. Navigate to the **Configuration** administration tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. Choose a **Default Resource Catalog** from the list provided.

   The catalog you select is offered to users when they add content to pages, page templates, and so on.

3. Click **Apply**.

43.3.3 Choosing a Default Navigation

Navigations enable users to easily get around your Portal Framework application and quickly access the information they need. For more information, see the "Developing a Navigation Model" chapter in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

Administrators can define the navigation used wherever there is a EL reference to `${navigationContext.defaultNavigation}`. This enables administrators to specify a default navigation model once and have it change throughout the system.

To choose a default navigation for the application:

1. Navigate to the **Configuration** administration tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. Choose a **Default Navigation** from the list provided.

   The navigation you select is applied wherever there is a EL reference to `${navigationContext.defaultNavigation}`.

   Note: If users hard-code navigation model references (for example, in a parameter to a task flow), the **Default Navigation** has no effect.

3. Click **Apply**.

43.3.4 Choosing a Default Skin

System administrators can customize the default appearance of a Portal Framework application by changing its skin. A skin changes the way the user interface appears, but does not change the application's behavior.

To choose a default skin:

1. Navigate to the **Configuration** administration tab.

See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. Choose a **Default Skin** from the list provided.

   The skin you select is applied to all the pages in your application.

   Note: The **Default Skin** has no effect on administration pages because the WebCenter Portal Administration Console uses an *internal skin* that does not change.

3. Click **Apply**.

43.3.5 Choosing the Default Base Resource URL

Developers can use EL expressions to dynamically generate the target URL for static resources. One way of doing this is to define a base URL preference to redirect resources to a desired server. With this option, EL expressions take a format that is illustrated by the following sample:

```
<af:image source="#{preferenceBean.baseResourceURL}/images/globe.png"/>
```

Administrators can configure the base URL at runtime in the application's administration console.

To choose the default base resource URL:

1. Navigate to the **Configuration** administration tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. In the **Base Resource URL** field, enter the details for the server to use for static resources, using the following format:

   protocol`://`*serverName*`:`*serverPortcontextPath*

   For example:

   `http://myserver.com:7101/myFolder`

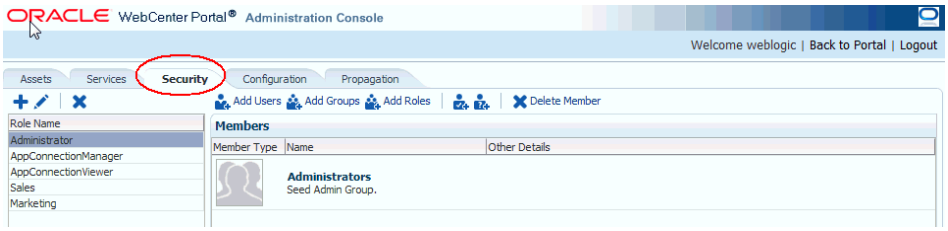
   Tip: You can also use an EL expression that resolves to a valid server. For example, the default setting is:

   ```
   #{request.scheme}://#{request.serverName}:#{request.serverPort}#{re
   quest.contextPath}
   ```

43.4 Managing Members and Roles for Portal Framework Applications

Users who are granted the `Administrator` role, can manage application members and roles through the WebCenter Portal Administration Console. From the **Security** tab (Figure 43–4), you can add members, define roles, grant and revoke permissions and roles, as well as remove members.

Figure 43–4 WebCenter Portal Administration Console - Security Tab



This section contains the following topics:

- Section 43.4.1, "Understanding Users"
- Section 43.4.2, "Understanding Application Roles and Permissions"
- Section 43.4.3, "Managing Users"
- Section 43.4.4, "Managing Application Roles and Permissions"

> **Note:** The **Security** tab displays the *Role Manager* task flow, which can be added independently to any Portal Framework application. For more information, see the "Using the Role Manager Task Flow" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

43.4.1 Understanding Users

Portal Framework application users each require a login account—provisioned directly from an existing identity store. Initially, only the system administrator (`weblogic` by default) can login and this default user has full administrative privileges through the `Administrator` role (see Table 43–1).

It is the system administrator's job to make individual users and user groups, who exist in the identity store, members of the application and to assign each member an appropriate application role. Default and custom application roles are described in Section 43.4.2.1, "Understanding Application Roles."

Alternatively, the system administrator may also choose to assign the application `Administrator` role to one or more other users and delegate this responsibility to others.

Table 43–1 Default Administrator for Portal Framework Applications

User	Description
System Administrator (`weblogic`)	Administrator for the entire Oracle WebCenter Portal domain. This user can manage any application within the domain.

43.4.2 Understanding Application Roles and Permissions

Application roles control the level of access a user has to assets and services in a Portal Framework application. This section describes application roles and permissions and includes the following topics:

- Section 43.4.2.1, "Understanding Application Roles"
- Section 43.4.2.2, "Understanding Application Permissions"

43.4.2.1 Understanding Application Roles

Application role assignment is the responsibility of the system administrator. Administrators can assign members a default application role or create additional, custom roles specific to their application. For more detail, see:

- Section 43.4.2.1.1, "Default Application Roles"
- Section 43.4.2.1.2, "Custom Application Roles"
- Section 43.4.2.2.1, "Application Permissions"
- Section 43.4.2.2.2, "Discussion Server Role Mapping"
- Section 43.4.2.2.3, "Understanding Enterprise Group Role Mapping"

Application roles and permissions defined within a Portal Framework application are stored in its *policy store* and, consequently, only apply to the application and do not imply any other permissions within the WebCenter Portal domain or other domains. Enterprise roles are different; enterprise roles are stored within the application's identity store and do not imply any permissions within the application.

43.4.2.1.1 Default Application Roles

Portal Framework applications provide several default application roles that cannot be deleted (Table 43–2).

Table 43–2 Default Roles for a Portal Framework Application

Application Role	Description
`Administrator`	Users with the `Administrator` role can set application-wide preferences, manage assets, configure the content repository, create polls, and register producers and external applications. Administrators can also manage users and roles for the application, and delegate or revoke privileges to/from other users. Initially, the `Administrators` enterprise group is the only member assigned full administrative privileges through the `Administrator` role. This means that any user in the `Administrators` group has full administrative privileges in the Portal Framework application.
`AppConnectionManager`	Users with this role can manage (*create*, *update*, and *delete*) portlet producers and external applications through corresponding task flows. Initially, only users with the `Administrator` role is a member of the `AppConnectionManager` role.
`AppConnectionViewer`	Users with this role can *view* portlet producers and external applications through corresponding task flows.. Initially, any user who is logged in (that is, has `authenticated-role`) is a member of the `AppConnectionViewer` role. No direct permissions are granted to the `AppConnectionViewer` role so everything granted to `authenticated-role` is available to members with this role.
`authenticated-role`	Authenticated users are granted the `authenticated-role`—a standard OPSS (Oracle Platform Security Services) application role.

Table 43–2 (Cont.) Default Roles for a Portal Framework Application

Application Role	Description
`anonymous-role`	Anyone who can access the application but is not logged in, is granted the `anonymous-role`—a standard OPSS (Oracle Platform Security Services) application role. Such users are anonymous, unidentified, and can see public content only.

43.4.2.1.2 Custom Application Roles

Custom application roles (sometimes known as user-defined roles) are specific to your Portal Framework application. When setting up the application, it is the administrator's job to identify which application roles are required, select suitable role names, and define the responsibilities of each role.

For example, an education environment might require roles such as `Teacher`, `Student`, and `Guest`. While roles such as `Finance`, `Sales`, `Human Resources`, and `Support` would be more appropriate for a corporate environment.

To learn how to set up applications roles for users, see Section 43.4.4.1, "Defining Application Roles."

43.4.2.2 Understanding Application Permissions

Every application role has specific, defined capabilities known as permissions. These permissions allow individuals to perform specific actions within the application. Note that no permission, inherits privileges from other permissions.

This section contains the following topics:

- Section 43.4.2.2.1, "Application Permissions"
- Section 43.4.2.2.2, "Discussion Server Role Mapping"
- Section 43.4.2.2.3, "Understanding Enterprise Group Role Mapping"

43.4.2.2.1 Application Permissions

Application permissions are categorized and listed in Table 43–3:

No permission, except for `Manage`, inherits privileges from other permissions.

Table 43–3 Application Permissions in Portal Framework Applications

Category	Application Permissions
Application	**Manage** - Enables access to all *Administration Console* pages: Assets, Services, Security, Configuration, Propagation. Through these pages, users can manage application security (users/roles), configure application-wide properties and services, manage application assets, create and manage pages, and propagate application changes. **Configure** - Enables users to view and perform operations on the Configuration tab. **Propagate** - Enables users to view and perform operations on the Propagation tab.

Table 43–3 (Cont.) Application Permissions in Portal Framework Applications

Category	Application Permissions
Task Flow Styles	**Create, Edit, and Delete** - Create, edit and delete task flow styles for the application using the WebCenter Portal Administration Console (Assets tab). **Create** - Create task flow styles for the application. **Edit** - Edit task flow styles. See Chapter 43.5, "Managing Assets for a Portal Framework Application."
Content Presenter Templates	**Create, Edit, and Delete** - Upload, edit and delete content presenter templates for the application using the WebCenter Portal Administration Console (Assets tab). **Create** - Upload content presenter templates for the application. **Edit** - Edit application-level content presenter templates. See Chapter 43.5, "Managing Assets for a Portal Framework Application."
Skins	**Create, Edit, and Delete** - Create, edit and delete skins using the WebCenter Portal Administration Console (Assets tab). **Create** - Create skins for the application. **Edit** - Edit skins. See Chapter 43.5, "Managing Assets for a Portal Framework Application."
Task Flows	**Create, Edit, and Delete** - Create, edit and delete task flows based on a task flow style using the WebCenter Portal Administration Console (Assets tab). **Create** - Create task flows for the application. **Edit** - Edit task flows. See Chapter 43.5, "Managing Assets for a Portal Framework Application."
Resource Catalogs	**Create, Edit, and Delete** - Create, edit and delete resource catalogs for the application using the WebCenter Portal Administration Console (Assets tab). **Create** - Create resource catalogs for the application. **Edit** - Edit resource catalogs. See Chapter 43.5, "Managing Assets for a Portal Framework Application."
Page Styles	**Create, Edit, and Delete** - Create, edit and delete page styles using the WebCenter Portal Administration Console (Assets tab). **Create** - Create page styles for the application. **Edit** - Edit page styles. See Chapter 43.5, "Managing Assets for a Portal Framework Application."
Data Controls	**Create, Edit, and Delete** - Create, edit and delete data controls for the application using the WebCenter Portal Administration Console (Assets tab). **Create** - Create data controls for the application. **Edit** - Edit data controls. See Chapter 43.5, "Managing Assets for a Portal Framework Application."

Table 43–3 (Cont.) Application Permissions in Portal Framework Applications

Category	Application Permissions
Navigations	**Create, Edit, and Delete** - Create, edit and delete navigations for the application using the WebCenter Portal Administration Console (Assets tab). **Create** - Create navigations for the application. **Edit** - Edit navigations. See Chapter 43.5, "Managing Assets for a Portal Framework Application."
Page Templates	**Create, Edit, and Delete** - Create, edit and delete page templates using the WebCenter Portal Administration Console (Assets tab). **Create** - Create page templates for the application. **Edit** - Edit page templates. See Chapter 43.5, "Managing Assets for a Portal Framework Application."
Pages	**Grant Page Access** - Manage page security. **Edit Pages** - Add or edit page content, rearrange content, and set page parameters and properties. **Customize Pages** - Customize pages for everyone. **Personalize Pages** - Personalize your view of pages by adding, editing, or removing content. **View** - View pages.
Links	**Create, and Delete** - Create and delete links between objects, and manage link permissions. **Delete** - Delete a link between two objects. **Create** - Create links between objects.
Lists	**Create, Edit, and Delete** - Create, edit, and delete lists and list data. **Create Lists** - Create lists. **Edit Lists** - Edit list column definitions. **Delete Lists** - Delete any list. **Edit List Data** - Add, edit, and delete list data. **View Lists** - View lists and list data.
People Connections	**Manage People Connections** -Manage application-wide settings for People Connection services. **Update People Connections Data** -Edit content associated with People Connection services. **Connect with People** -Share content associated with People Connection services with others.

43.4.2.2.2 Discussion Server Role Mapping

Some services that need access to "remote" (back-end) resources also require role-mapping based authorization. That is, the roles that allow users to work with the discussions in a Portal Framework application, must be mapped to corresponding roles on the back-end discussions server.

A Portal Framework application uses *application roles* to manage user permissions within the application. On the discussions server, a different set of roles and permissions apply.

A Portal Framework application user who is assigned the `Discussions-Create Edit Delete` permission is automatically added as a discussions server user and assigned the `Administrator` role (on the discussions server) with `Category Admin` permissions. In Portal Framework applications, the `Administrator` role is granted the `Discussions-Create Edit Delete` permission by default as shown in Table 43–4.

Table 43–4 Discussions Server Roles and Permissions

Discussion Server Role	Discussion Server Permissions	Portal Framework Application Equivalent Application Permission
Administrator	Category Admin	`Discussions-Create, Edit, and Delete` Create, read, update and delete sub categories, forums and topics inside the category for which permissions are granted.

43.4.2.2.3 Understanding Enterprise Group Role Mapping

You can assign individual users or multiple users in the same enterprise group to Portal Framework application roles. Subsequent enterprise group updates in the back-end identity store are then automatically reflected in the Portal Framework application. Initially, when you assign an enterprise group to a Portal Framework application role, everyone in the enterprise group is granted that role. If someone moves out of the group, the role is revoked. If someone joins the group, they are granted the role.

For a Portal Framework application to properly maintain enterprise group-to-role mappings, back-end servers, such as the discussions server and content server, must support enterprise groups too. WebCenter Portal's discussions server and WebCenter Content's Content Server versions provided with this release both support enterprise groups but previous versions may not.

Note: Adding users to groups that have been assigned a role is not dynamic. That is, the user may not immediately have access to assets based on the role. To work around this, you can either clear the cache using the `refreshGroupSpaceCache` WLST command as shown in the example below:

```
refreshGroupSpaceCache(appName='webcenter', spaceNames='',
syncMode=true, updateType='all', cleanCache=false)
```

or just wait for the cache to refresh.

43.4.3 Managing Users

Administrators must ensure that all application users have appropriate permissions. To get permissions, users must be granted membership to the application through an appropriate application role.

This section tells you how to add members and assign roles. It contains the following topics:

- Section 43.4.3.1, "Adding Members to Application Roles"

- Section 43.4.3.2, "Assigning a User to a Different Role"
- Section 43.4.3.3, "Giving a User Administrative Privileges"
- Section 43.4.3.4, "Revoking Application Roles"
- Section 43.4.3.5, "Adding or Removing Users"

43.4.3.1 Adding Members to Application Roles

You can grant membership to individual users or multiple users in the same enterprise group through the Security tab. Any user or group defined in the identity store is eligible for membership. See Section 31.3, "Adding Users to the Embedded LDAP Identity Store."

Updates in your back-end identity store, such as new users or someone leaving an enterprise group, are automatically reflected in the Portal Framework application. Initially, when you assign an enterprise group to an application role, everyone in the group is granted that role. If someone moves out of the group, the role is revoked. If someone joins the group, they are granted the role.

> **Note:** For a Portal Framework application to properly maintain enterprise group-to-role mappings, back-end servers, such as the discussions server and content server, must support enterprise groups too. WebCenter Portal's discussion server and Oracle WebCenter Content provided with WebCenter Portal 11.1.1.2.0 and later support enterprise groups but earlier versions may not.

To grant user membership through an appropriate application role:

1. Navigate to the **Security** administration tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. In the **Role Name** pane, select the role to assign to the user.

   Notice that the current list of members assigned to the role you select are listed on the right (in the **Members** pane).

   Only choose **Administrator** to assign full, administrative privileges for your application. If the role you want is not listed, create a new role that meets your requirements (see Section 43.4.4.1, "Defining Application Roles").

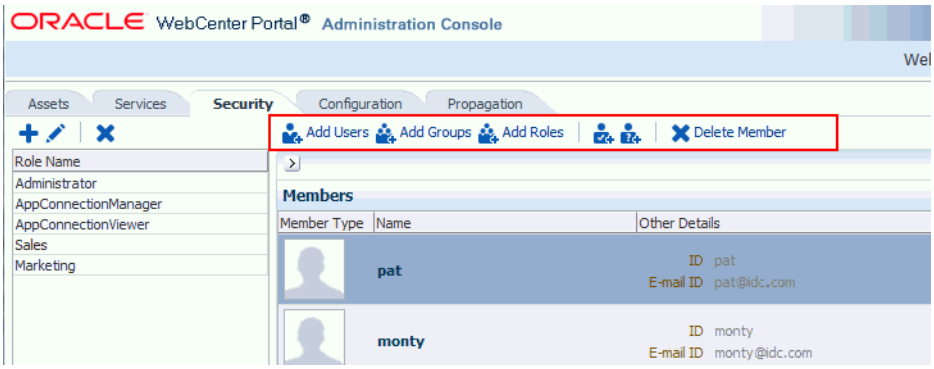
3. Click **Add Users**, **Add Groups**, or **Add Roles** (Figure 43–5):

   Add Users - click to grant membership to individual users

   Add Groups - click to grant membership to everyone in a user group

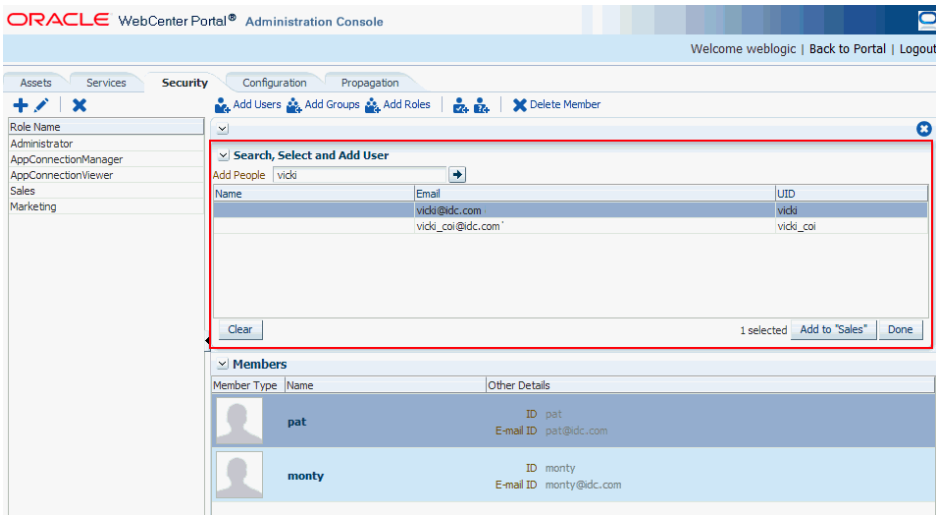
   Add Roles - click to grant membership to everyone assigned to a particular application role

Figure 43–5 WebCenter Portal Administration Console - Add Members



4. If you know the exact name of the user, group, or application role, enter the name in the search box (**Add People**, **Add Group** or **Add Role**) and click the arrow icon. If you are not sure of the name you can search your identity store using part of the name as shown below (see Figure 43–6).

Figure 43–6 Add Users Pane - People Search



5. Select the user, group, or role to whom you want to grant the selected role, and click **Add to**. Use the Ctrl key to select multiple names.

> **Note:** If the user, group, or role already has the role, directly or indirectly, the role is not granted; cyclic role assignments are not allowed.

The new members display in the Members pane.

6. Click **Done** when finished.

43.4.3.2 Assigning a User to a Different Role

From time to time, a user's role in a Portal Framework application may change. For example, a user may move out of sales into the finance department and in this instance, the user's role assignment may change from *Sales* to *Finance*. To change a user's role, first revoke membership for the user's previous role, then add the user to

the new role. For more information see Section 43.4.3.4, "Revoking Application Roles" and Section 43.4.3.1, "Adding Members to Application Roles."

> **Note:** You cannot modify your own role or the system administrator's role. See Section 43.4.2.1, "Understanding Application Roles."

43.4.3.3 Giving a User Administrative Privileges

It's easy to give a user full, administrative privileges for your Portal Framework application through the `Administrator` role. Administrators have the highest privilege level and can view and modify anything in the application so take care when assigning the `Administrator` role.

Most administrative tasks, such as managing users and roles, are exclusive to the `Administrator` role. See Section 43.4.2.1.1, "Default Application Roles."

43.4.3.4 Revoking Application Roles

It's easy to revoke application role assignments that no longer apply. Note, however, that revoking all of a user's application roles does not remove that user from the identity store.

> **Note:** You cannot revoke your own role assignments or the system administrator's role. See Section 43.4.2, "Understanding Application Roles and Permissions."

To revoke application roles:

1. Navigate to the **Security** administration tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. Select the role from the **Role Name** pane.
3. Select a user, group, or application role from the Members list.
4. Click **Delete Member**.
5. When prompted, click **Delete** to revoke the role for the user.

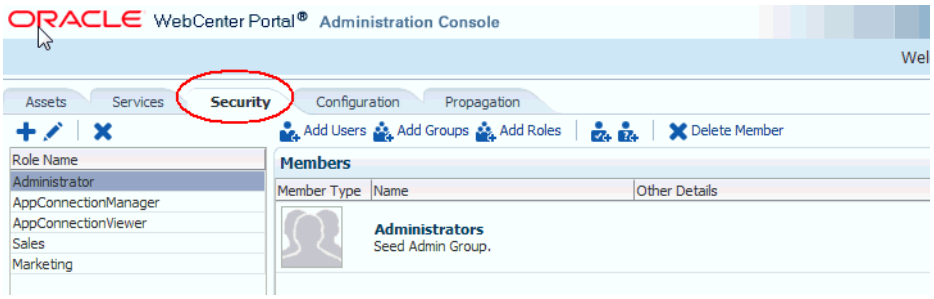
43.4.3.5 Adding or Removing Users

Administrators cannot add new user data directly to the Portal Framework application's identity store or remove user credentials. Identity store management is the responsibility of the systems administrator and takes place through the WLS Administration Console or directly into embedded LDAP identity stores using LDAP commands. See Section 31.3, "Adding Users to the Embedded LDAP Identity Store."

43.4.4 Managing Application Roles and Permissions

Portal Framework applications use application roles to manage permissions for users within the application. This section tells you how to manage application roles, and their permissions from the Security administration page (Figure 43–7).

Figure 43–7 WebCenter Portal Administration Console - Security Tab



Portal Framework applications provide several default application roles. You cannot delete default application roles but you can modify the default permission assignments for each role. For more information, see Section 43.4.2, "Understanding Application Roles and Permissions."

This section contains the following topics:

- Section 43.4.4.1, "Defining Application Roles"
- Section 43.4.4.2, "Modifying Application Role Permissions"
- Section 43.4.4.3, "Granting or Removing Roles for Unauthenticated Users"
- Section 43.4.4.4, "Granting Roles to All Authenticated Users"
- Section 43.4.4.5, "Deleting Application Roles"

43.4.4.1 Defining Application Roles

Use roles to characterize groups of application users and determine what they can see and do within their Portal Framework application.

When defining application roles, use self-descriptive role names and try to keep the role policy as simple as possible. Choose as few roles as you can, while maintaining an effective policy.

Take care to assign appropriate access rights when assigning permissions for new roles. Do not allow users to perform more actions than are necessary for the role, but at the same time, try not to inadvertently restrict them from activities they must perform. In some cases, users might fall into multiple roles.

To define a new application role:

1. Navigate to the **Security** administration tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications".

2. Click the **Add** icon (plus) above the Role Name pane.

   The Create Role dialog displays (Figure 43–8).

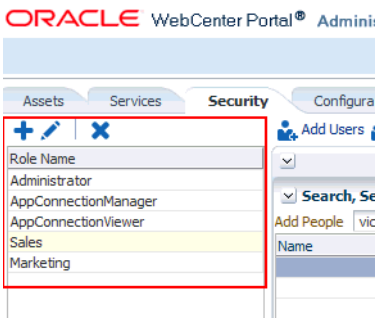
Figure 43–8 WebCenter Portal Administration Console - Create Role Dialog

3. Enter a name and description for the new role, and click **Create Role**.

   Ensure the role name is self-descriptive. Make it as obvious as possible which users should belong to which roles. Role names can contain alphanumeric characters, blank spaces, @, and underscores.

   The new role is listed in the Roles pane (Figure 43–9).

Figure 43–9 WebCenter Portal Administration Console - Roles Pane



4. Continue by defining the user permissions for the role as described in Section 43.4.4.2, "Modifying Application Role Permissions."

43.4.4.2 Modifying Application Role Permissions

Permissions should always be set after creating a new role, but administrators can modify the permissions associated with an application role at any time.

Application role permissions allow individuals to perform specific actions within their application. Application permissions are described in Section 43.4.2, "Understanding Application Roles and Permissions."

To change the permissions assigned to a role:

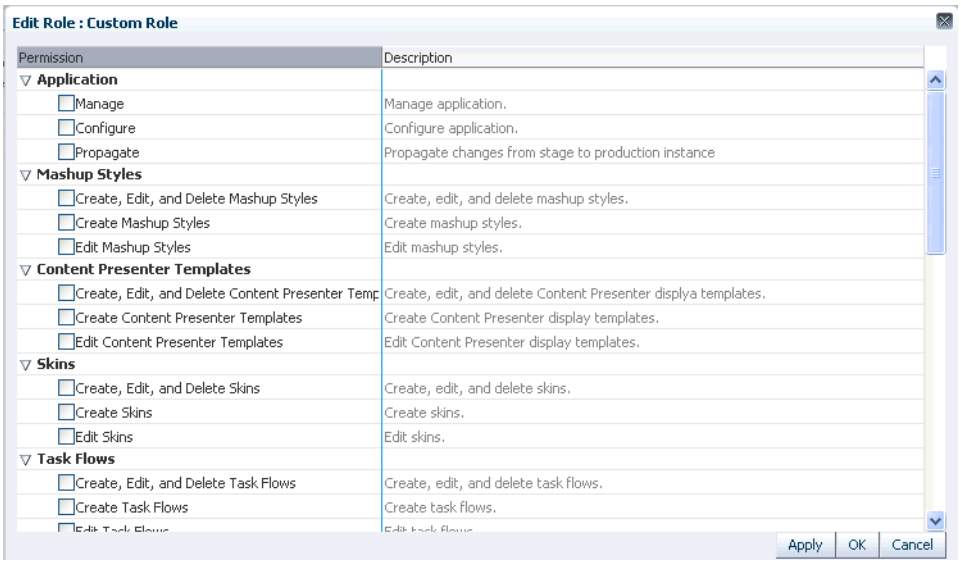
1. Navigate to the **Security** administration tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. Select the role for which to view or modify permissions, and click the **Edit** icon (pencil).

   The Edit Role dialog displays with the current permissions for the role (Figure 43–10).

Figure 43–10 WebCenter Portal Administration Console - Edit Role Dialog



3. Select or clear permissions check boxes to enable or disable permissions for the role.
4. Click **Apply** to save or **OK** to save and exit.

The new permissions are effective immediately.

43.4.4.3 Granting or Removing Roles for Unauthenticated Users

Anyone who is not logged in to a Portal Framework application assumes the `anonymous-role`. Initially, users with the `anonymous-role` have no privileges and only see public application pages, such as the login or landing page, and also content that individual users choose to make public.

> **Caution:** Take care when granting privileges to the `anonymous-role`. Avoid granting administrative privileges, or any permission that might be considered unnecessary. For security reasons, Oracle recommends that you limit what anonymous users can see and do in your application. If you have no public or anonymous access to your application, any grants that are currently given to `anonymous-role` should be removed or moved to `authenticated-role`.

To grant application roles to the public:

1. Navigate to the **Security** administration tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

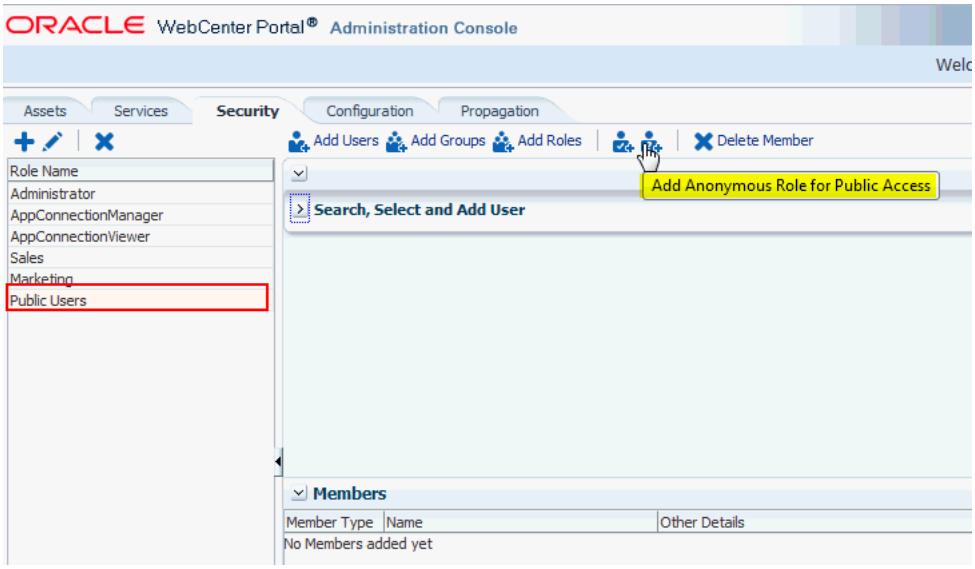
2. If you have not done so already, define a role that grants permissions suitable for unauthenticated users.

   See Section 43.4.4.1, "Defining Application Roles."

3. Select the role that defines privileges suitable for unauthenticated users.

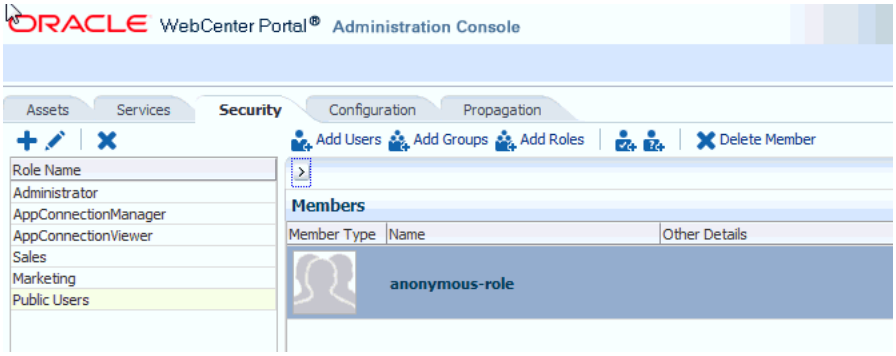
4. Click **Add Anonymous Role for Public Access** (Figure 43–11). (If the `anonymous-role` has already been added and you want to remove it, click **Delete**.)

Figure 43–11 WebCenter Portal Administration Console - Add Public Access



The anonymous-role is added to the members list (Figure 43–12).

Figure 43–12 WebCenter Portal Administration Console - Add Public Access



43.4.4.4 Granting Roles to All Authenticated Users

Anyone who is logged in to a Portal Framework application assumes the `authenticated-role`.

Other important notes:

- Since all authenticated users are also under `anonymous-role`, all grants given to `anonymous-role` automatically accrue to authenticated users.
- Custom application roles all inherit permissions from the `authenticated-role`.

43.4.4.5 Deleting Application Roles

When an application role is no longer required you should remove it from your application. This helps maintain a valid role list, and prevents inappropriate role

assignment. You cannot, however, delete a role that is granted to you, directly or indirectly.

Application roles are deleted even when users are still assigned to them.

> **Note:** Default roles cannot be deleted: `Administrator`, `AppConnectionManager`, `AppConnectionViewer`, `authenticated-role`, `anonymous-role`. See Section 43.4.2.1.1, "Default Application Roles."

As you cannot delete *default roles*, application users can log in through the `authenticated-role`, even when all other application roles are revoked.

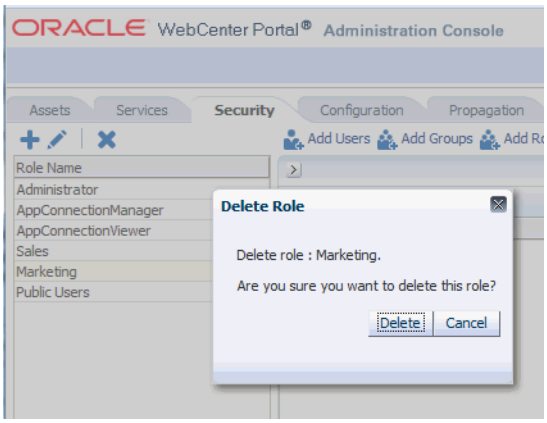
To delete an application role:

1. Navigate to the **Security** administration tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. Select the role to delete from the list of roles and click the **Delete** icon (**x**).

Figure 43–13 Deleting an Application Role



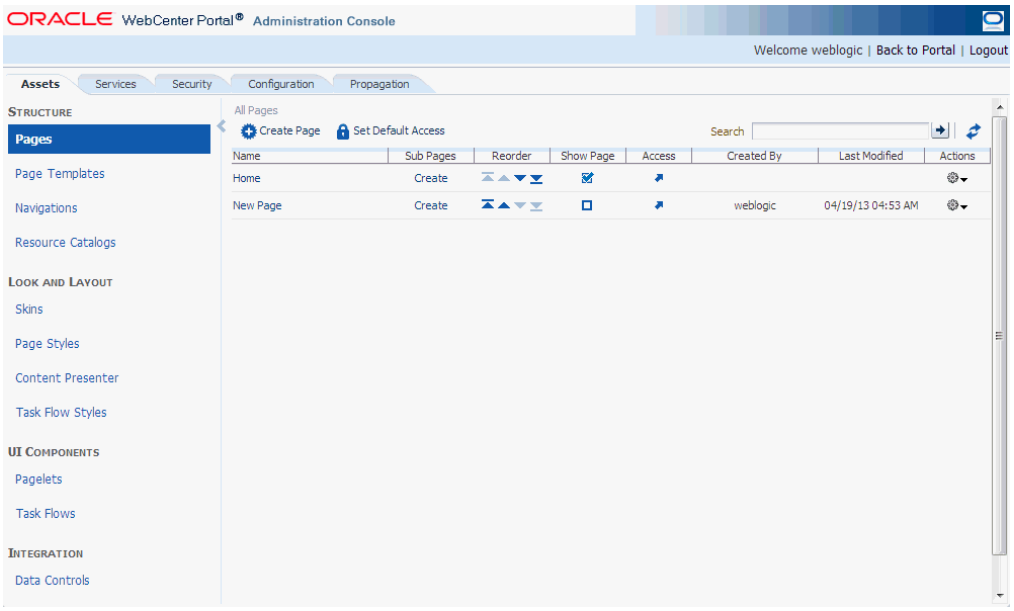
3. When prompted, click **Delete** to confirm that you want to delete the role.

   The role is removed from the table. Users assigned to only this role can still log in through the `authenticated-user` role.

43.5 Managing Assets for a Portal Framework Application

By using the Assets page of the administration console, you can manage your application assets—pages, page templates, navigations, resource catalogs, skins, page styles, content presenter display templates, task flow styles, data controls, and task flows (Figure 43–14). You can perform tasks such as create, edit, copy, publish, upload, and download your application assets. For information about various assets, see the "Introduction to Portal Resources" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

Figure 43–14 Administration Console - Assets Page



This section includes the following topics:

- Section 43.5.1, "Working with Pages"
- Section 43.5.2, "Creating an Asset"
- Section 43.5.3, "Copying an Asset"
- Section 43.5.4, "Editing Assets"
- Section 43.5.5, "Setting Properties on an Asset"
- Section 43.5.6, "Showing or Hiding an Asset"
- Section 43.5.7, "Setting Asset Security"
- Section 43.5.8, "Uploading and Downloading an Asset"
- Section 43.5.9, "Previewing an Asset"
- Section 43.5.10, "Deleting an Asset"

43.5.1 Working with Pages

At runtime, you can create and manage application pages.

This section includes the following topics:

- Section 43.5.1.1, "Creating a Page"
- Section 43.5.1.2, "Creating a Subpage"
- Section 43.5.1.3, "Setting Page Access"
- Section 43.5.1.4, "Reordering a Page"
- Section 43.5.1.5, "Moving a Page in the Page Hierarchy"
- Section 43.5.1.6, "Renaming a Page"

For information about editing, copying, or deleting a page, refer to the generic resource procedures documented later in this chapter.

43.5.1.1 Creating a Page

To create an application page at runtime:

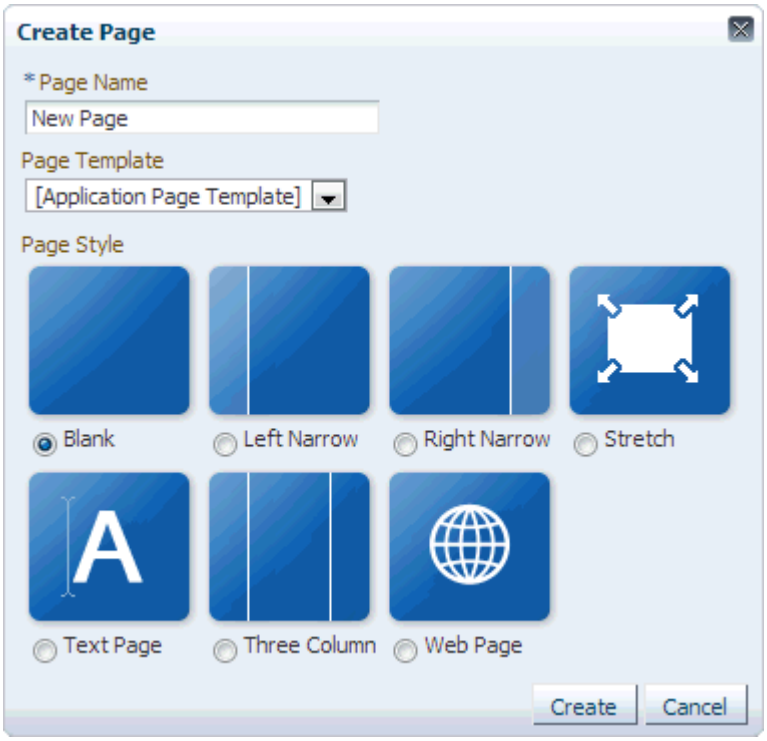
1. Open the **Assets** tab in the administration console.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. In the navigation panel on the left, click **Pages**.
3. On the menu bar, click **Create Page**.
4. In the Create Page dialog (Figure 43–15), in the **Page Name** field, enter the name of the page.
5. From the **Page Template** list, select the page template on which you want to base your page.
6. From the **Page Style** list, select the style you want to use for your page.

   For information about page styles, see the "Introduction to Page Styles" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

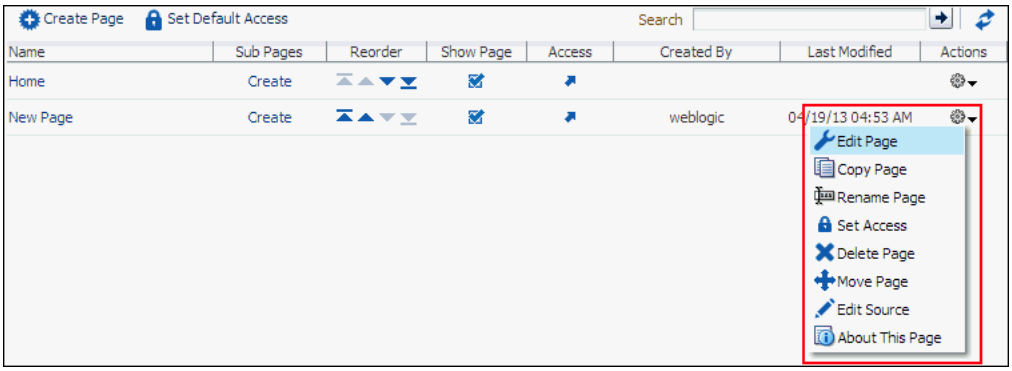
Figure 43–15 Create Page Dialog



7. Click **Create**.

   The newly created page is listed on the Assets page. You can mange the page by using the options available on the **Actions** menu of the page (Figure 43–16).

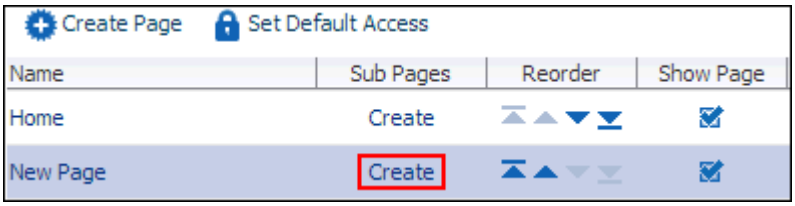
Figure 43–16 Actions Menu of a Page



43.5.1.2 Creating a Subpage

The procedure to create a subpage is similar to creating a main page as described in Section 43.5.1.1, "Creating a Page." On the **Assets** page, you need to click **Create** in the Sub Pages column for the page under which you want to create the subpage (Figure 43–17). By default, subpages inherit security from their parent page.

Figure 43–17 Create Link for Creating a Subpage

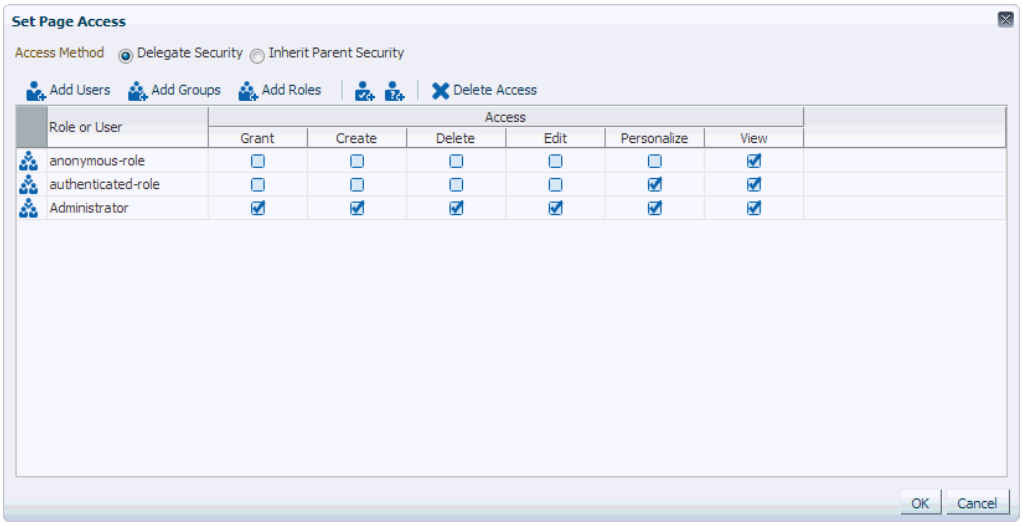


43.5.1.3 Setting Page Access

All your application pages reside under the root node, and by default, inherit the permissions defined for the root node. You can define custom permissions for the root node. Further, you can override the default root node permissions, and specify custom permissions for individual pages. By default, subpages inherit security from their parent page.

While setting page access, you can choose either of these access methods: **Delegate Security** or **Inherit Parent Security** (Figure 43–18).

Figure 43–18 Set Page Access Dialog



Choose:

- **Delegate Security** - to define who may interact with a page. When you select this option, all permissions that a page currently inherits from its parent are displayed in the dialog. This effectively is the current security policy applicable on the page. You can further refine the grants by adding new grants or removing the existing ones.
- **Inherit Parent Security** - to inherit permissions defined for a page's parent node. When you set this option, any security permissions defined on the page are deleted, and the parent node's security settings take effect.

You can define the following page permission actions:

- **Manage** - grants all other page permissions and is typically used for specifying a super administrator type of access. This permission action is available only when defining access on the root node.
- **Grant** - allows a user to further grant the access that they already have to other users, groups, or roles. For example, if a user has the Grant and Edit permissions, the user can further grant these two actions only; she cannot grant any other permission, like Delete or Personalize.
- **Create** - allows a user to create subpages under the current page.
- **Delete** - allows a user to delete a page along with all its subpages.
- **Edit** - allows a user to edit a page.
- **Personalize** - allows a user to personalize a page.
- **View** - allows a user to view a page.

All permissions follow one of the two permission models - delegation or containment. Delegation is when an entity has been granted a particular permission on a page, and this is all that is used to evaluate whether the entity has the said permission action or not. All permission actions other than View fall in this category. A permission is of containment type if the permission is granted to an entity on a page, as well as all the nodes up in the hierarchy where security is defined. Only the View permission action falls in this category.

So, to be able to view a page, you need to have the View permission action on the specified page and all nodes up in the hierarchy to the root node. If you do not have the View permission on an intermediate node in the page hierarchy, you cannot view the specified page. For other permission actions (delete, edit, and the like), you just need that particular permission on the page.

This section includes the following topics:

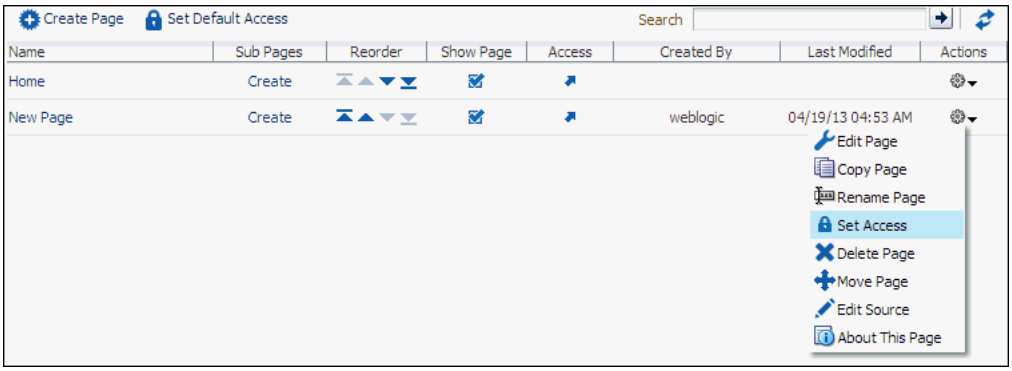
- Section 43.5.1.3.1, "Setting Permissions on an Individual Page"
- Section 43.5.1.3.2, "Setting Permissions on the Root Node"

43.5.1.3.1 Setting Permissions on an Individual Page

To set permissions on a specific page:

1. Navigate to the **Assets** page, as described in Section 43.2, "Accessing the Administration Console for Portal Framework Applications."
2. In the left navigation panel, click **Pages**.
3. From the list of pages, open the **Actions** menu for the required page, then select **Set Access**.

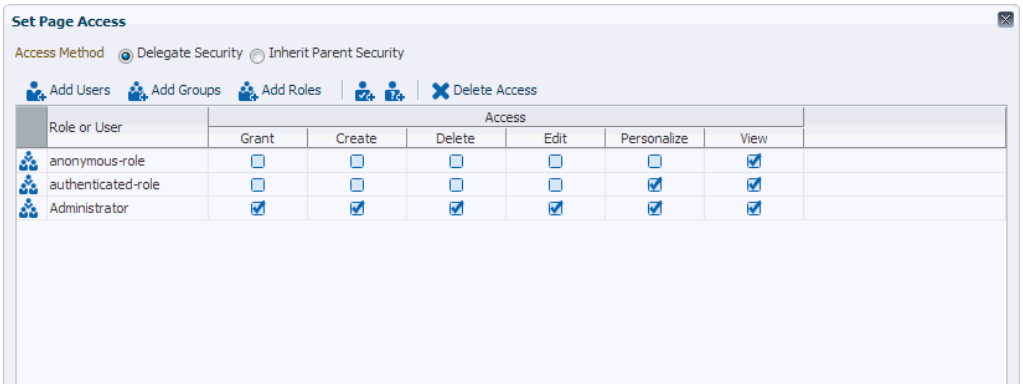
Figure 43–19 Setting Page Access



4. In the Set Page Access dialog, select an access method (Figure 43–20):
   - **Delegate Security**—Select this method to define who may manage and update this page node, and all its children in the hierarchy that have not overridden security. Selecting this method shows the default permissions available. You can further refine the permissions by adding new permissions or removing the existing ones.

     If you select this option, proceed to step 5.
   - **Inherit Parent Security**—Select this method to inherit permissions defined for the page's parent node. If you select this method, click **OK** to save your changes and exit the dialog.

Figure 43–20 Setting Page Access

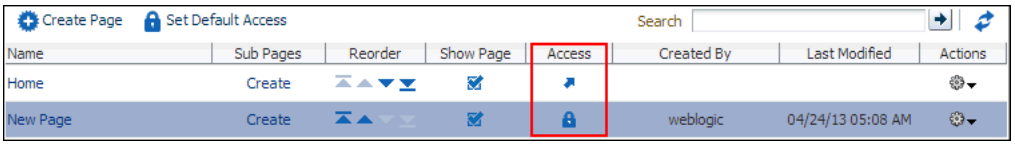


5. If you selected **Delegate Security**, specify the user, group, or role to which you want to grant access to the page.
   - Click **Add Users** to search for and select individual users included in your identity store. Select the required user(s), and click **OK**.
   - Click **Add Groups** to search for and select groups of users included in your identity store. Select the required group(s), and click **OK**.
   - Click **Add Roles** to search for and select the application roles to which you want to grant access to the page. Select the required role(s), and click **OK**.
6. For each user, group, or role listed in the **Role or User** column, specify the level of access you want to grant. You can grant any of the following permissions: Grant, Create, Delete, Edit, Personalize, and View.

   See Section 43.5.1.3, "Setting Page Access" for a description of each permission.
7. Optionally, you can click the **Add Authenticated Role for Logged in User Access** icon to add **authenticated-role** and define the access for all authenticated users. You can click the **Add Anonymous Role for Public Access** icon to add **anonymous-role** and grant the required permissions to all users.
8. If you want to revoke permissions from any user, group, or role, select that entity and click **Delete Access**.
9. Click **OK** to save the security settings for the page.

   The lock icon in the Access column for a page signifies that **Delegate Security** access method has been set for the page; the arrow icon signifies that **Inherit Parent Security** access method has been set for the page (Figure 43–21).

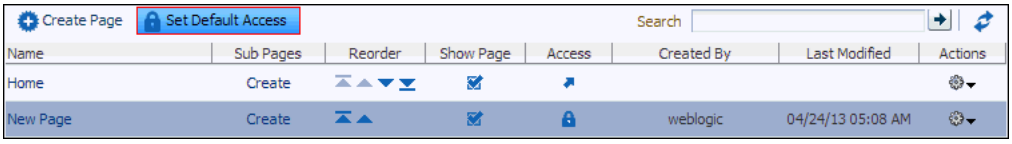
Figure 43–21 Access Methods Specified for Pages



43.5.1.3.2 Setting Permissions on the Root Node All page and subpage nodes that do not have security overridden, derive their access settings from the root node. You can define permissions for the root node by using the **Set Default Access** option on the **Pages** page on the **Assets** tab (Figure 43–22). The rest of the procedure is same as that for setting permissions for a specific page. For information, see Section 43.5.1.3.1, "Setting Permissions on an Individual Page."

When you set permissions for the root node, in addition to other permissions, you can also set the Manage permission. A user with this permission has complete access on the entire page hierarchy irrespective of the settings on individual pages.

Figure 43–22 Setting Access for the Root Node



43.5.1.4 Reordering a Page

You can reorder your pages. This order is used when you include pages in the navigation model through a Page Query.

To change the order of a page:

1. Navigate to the **Assets** page in the administration console.

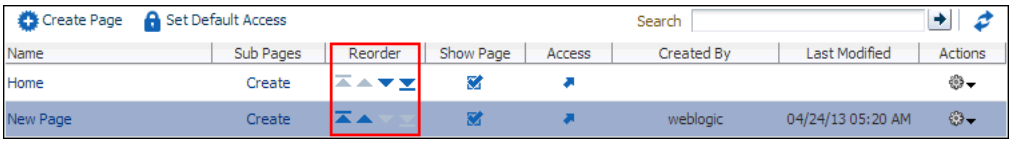
   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. In the left navigation panel, click **Pages**.

3. Drag the page to the required location in the page hierarchy.

   To reorder pages, you can also use the icons displayed in the Reorder column (Figure 43–23). Use the icons to move your page to the top of the list, bottom of the list, before the preceding page, or after the page displayed next in the list.

Figure 43–23 Reordering a Page



43.5.1.5 Moving a Page in the Page Hierarchy

You can change the level at which a page appears in the page hierarchy—you can move a page to appear as a subpage, appear at the root level, or appear as a parent page.

To move a page in the page hierarchy:

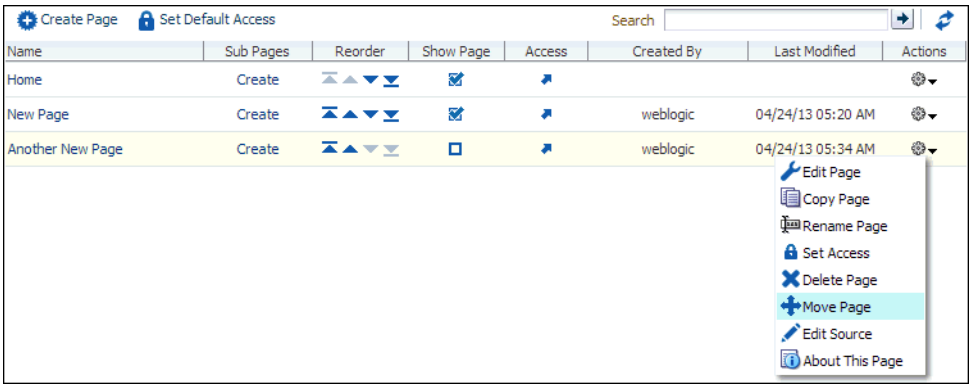
1. Navigate to the **Assets** page in the administration console.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. In the left navigation panel, click **Pages**.

3. Open the **Action** menu for the page that you want to move, and choose **Move Page** (Figure 43–24).

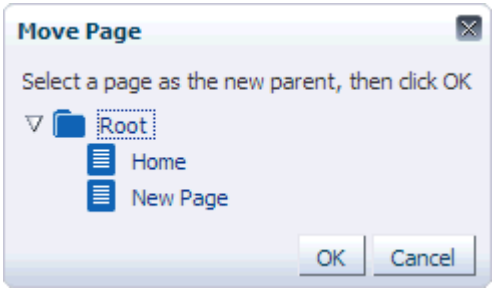
Figure 43–24 Actions Menu of a Page



4. In the Move Page dialog, select the level at which you want to move the page in the page hierarchy.

   For example, if you want your page to appear as a subpage of **MyPage**, click **MyPage** (Figure 43–25).

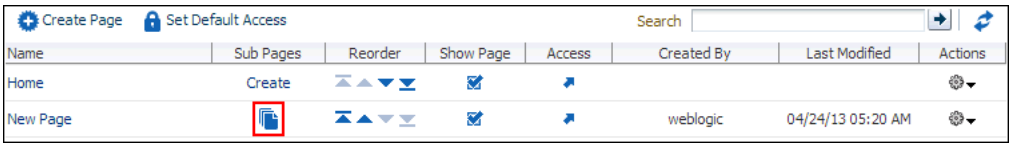
Figure 43–25 Moving a Page



5. Click **OK**.

   In Figure 43–26, the subpage icon in the **Sub Pages** column represents that **MyPage** now contains a subpage.

Figure 43–26 A Page Updated in the Page Hierarchy



43.5.1.6 Renaming a Page

To rename a page:

1. Navigate to the **Assets** page in the administration console, as described in Section 43.2, "Accessing the Administration Console for Portal Framework Applications."
2. In the left navigation panel, click **Page**.
3. Open the **Action** menu for the page you want to rename, and choose **Rename Page**.

4. In the Rename Page dialog, enter the desired name.
5. Click **OK**.

43.5.2 Creating an Asset

Even after your application has been deployed, as an administrator, you may need to constantly update it to meet your organization's requirements. Portal Framework applications enable you to create and edit assets at runtime, without requiring you to redeploy your application.

Notes: You cannot create Content Presenter display templates at runtime. For more information, see the "Creating Content Presenter Display Templates" chapter in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

To create an asset:

Notes: The procedure for creating a page, pagelet, or data control is different than other assets:

- For information about creating a page, see Section 43.5.1.1, "Creating a Page."
- For information about creating a pagelet, see "Creating Pagelets with Oracle WebCenter Portal's Pagelet Producer" in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.
- For information about creating a data control see the "Creating Data Controls" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*. The procedure to create a data control at runtime in a Portal Framework application is similar to that in WebCenter Portal.
- You can create page styles and task flow styles at runtime only by copying an existing style. For more information, see Section 43.5.3, "Copying an Asset."

---

1. Navigate to the **Assets** page in the administration console.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."
2. In the left navigation panel, select the category of the asset you want to create.
3. On the menu bar, click **Create**.
4. In the **Create** dialog, in the **Name** field, enter the name of the asset.
5. In the **Description** field, enter a description of the asset.
6. From the **Copy From** list, select the existing asset that you want to extend for creating a new asset.

Note: The **Copy from** list is available for page templates, navigations, resource catalogs, and skins. It is not available for task flows.

7. (For task flows only) Select the **Task Flow Style** that you want to use for the new asset.
8. Click **Create**.

   The newly created asset is listed on the **Assets** page. The empty check box next to an asset indicates that it is not yet published and hence not available to users for use. For information about publishing assets, see Section 43.5.6, "Showing or Hiding an Asset."

43.5.3 Copying an Asset

You can create a copy of an asset. This feature is useful when you want to create a backup of an asset, update an asset while keeping the original in use, or use an existing asset as the starting point for creating a new asset. When you create a copy of an resource, the copy is marked as hidden.

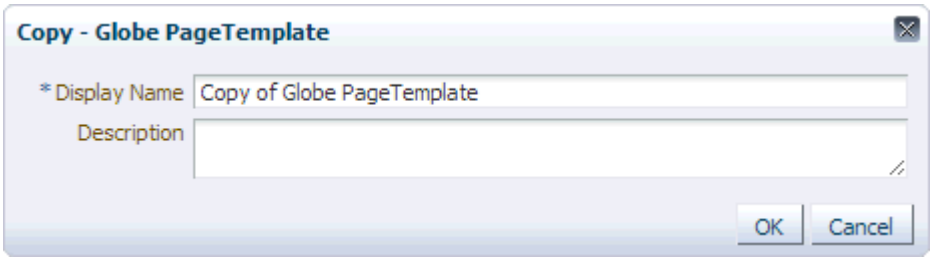
To make a copy of an asset:

1. Navigate to the **Assets** page in the administration console.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."
2. In the left navigation panel, select the desired type of asset.
3. From the list of assets, select the asset you want to copy.
4. From the **Actions** menu, choose **Copy**.

   Note: To make a copy of a page, select **Copy Page** from the **Actions** menu of the page. In the Copy Page dialog, specify the page name and click **OK**.
5. In the Copy dialog, in the **Display Name** field, enter a name for the copy (Figure 43–27).

Figure 43–27 Copying an Asset



6. In the **Description** field, enter a description of the new asset.
7. Click **OK**.

43.5.4 Editing Assets

At runtime, you can perform two types of asset editing:

- **Edit** - provides a way to edit an asset either with Composer or in an Edit dialog.
- **Edit Source** - enables you to work with the source code of an asset.

43.5.4.1 Editing an Asset Using the Edit Option

To edit an asset at runtime:

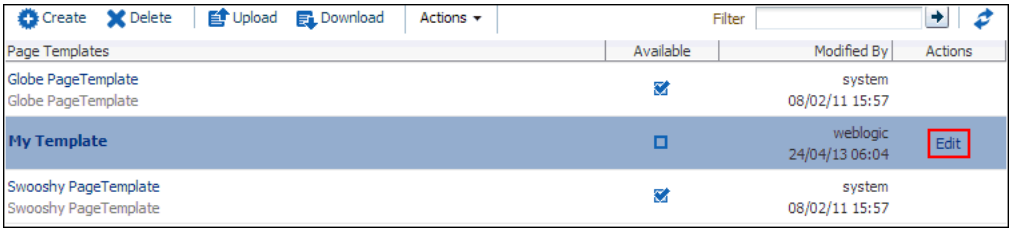
1. Navigate to the **Assets** page in the administration console.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. In the left navigation panel, select the desired type of asset.
3. From the list of assets, select the asset you want to edit.
4. In the Actions column, click **Edit** (Figure 43–28).

> **Note:** To edit a page, select the **Edit Page** option from the **Actions** menu of the page.

Figure 43–28 The Edit Option for Assets



5. Edit the asset as desired.

The properties that you can edit vary from asset to asset. For information about the properties of an asset that can be edited, refer to the relevant asset chapter listed in the "Editing an Asset Using the Edit Option" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*. The procedure to edit an asset at runtime in a Portal Framework application is similar to that in WebCenter Portal.

43.5.4.2 Editing the Source Code of an Asset

To get more control over asset editing at runtime, you can edit the underlying source code of any custom asset, except data controls and pages.

To edit the source code of an asset:

1. Navigate to the **Assets** page in the administration console.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

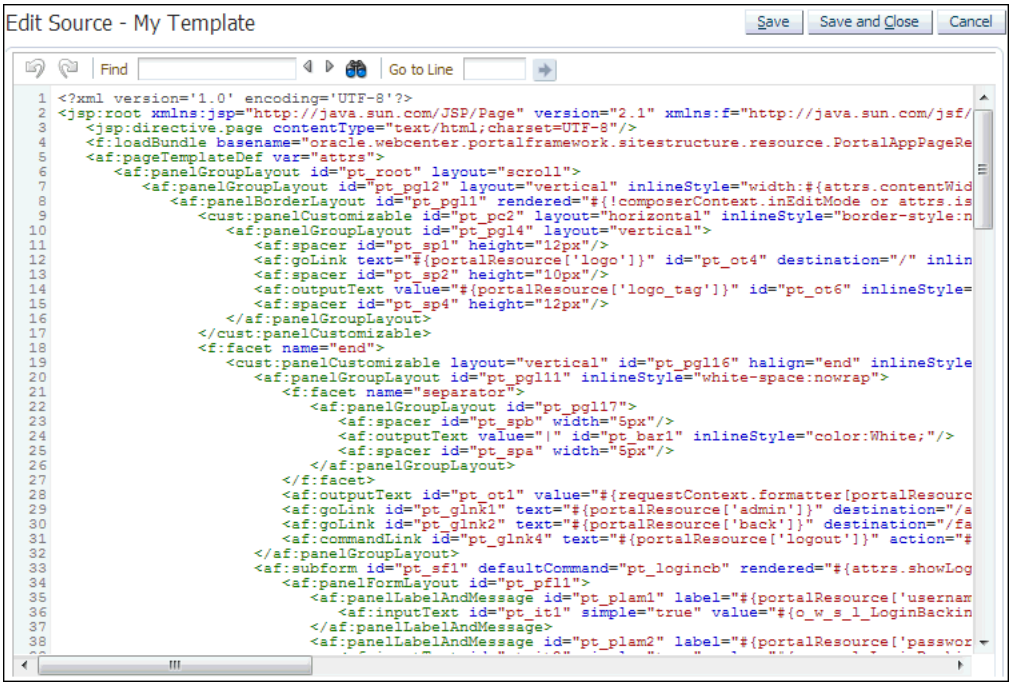
2. In the left navigation panel, select the desired type of asset.
3. From the list of assets, select the asset you want to edit.
4. From the **Actions** menu, choose **Edit Source**.

   The Edit Source dialog displays the asset definition.

5. Edit the code as required (Figure 43–29).

   The XML syntax in the code is validated and an error message is displayed if you miss any tags or add them incorrectly. Validation is not performed for non-XML files, such as a CSS file.

Figure 43–29 Editing the Source of an Asset



6. When you are done, click **Save and Close**.

43.5.5 Setting Properties on an Asset

Each asset has certain associated properties that define its display properties and attributes. Authorized users can edit these properties by using the Edit Properties dialog. For information about the properties displayed in the Edit Properties dialog, see the "Viewing Information About an Asset" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

This section describes how to access the Edit Properties dialog and set asset properties. It includes the following topics:

> **Note:** The properties described in this section are not applicable to a page asset.

- Section 43.5.5.1, "Accessing the Edit Properties Dialog of an Asset"
- Section 43.5.5.2, "Editing the Name or Description of an Asset"
- Section 43.5.5.3, "Associating an Icon with an Asset"
- Section 43.5.5.4, "Categorizing an Asset"
- Section 43.5.5.5, "Setting Asset Attributes"

43.5.5.1 Accessing the Edit Properties Dialog of an Asset

To access the Edit Properties dialog of an asset:

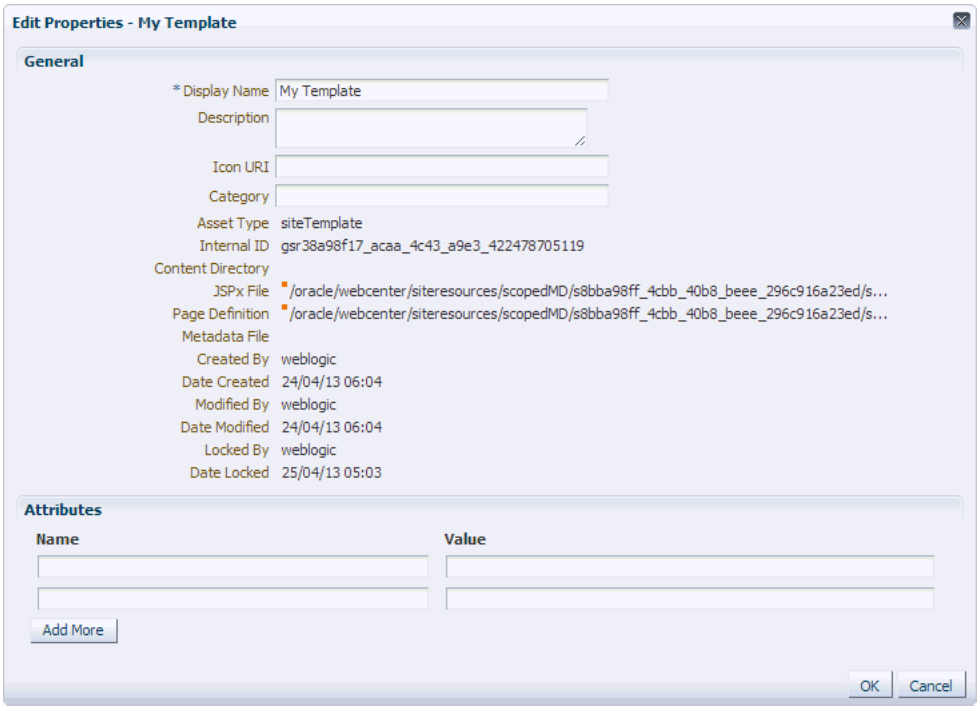
1. Navigate to the **Assets** page in the administration console.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. In the left navigation panel, select the desired type of asset.
3. From the list of assets, select the asset you want to edit.
4. From the **Actions** menu, choose **Edit Properties**.

   The Edit Properties dialog opens (Figure 43–30).

Figure 43–30 Edit Properties Dialog of an Asset



5. Edit the properties as desired. For more information see:
   - Section 43.5.5.2, "Editing the Name or Description of an Asset"
   - Section 43.5.5.3, "Associating an Icon with an Asset"
   - Section 43.5.5.4, "Categorizing an Asset"
   - Section 43.5.5.5, "Setting Asset Attributes"
6. When you are done, click **OK**.

43.5.5.2 Editing the Name or Description of an Asset

Assets are sorted on the Assets page according to their display names. To maintain a well-organized set of assets, consider developing a standard naming scheme and method of description. This is not a required step, but it may be useful in identifying and clarifying your intended purpose for a given asset.

To edit the name or description of an asset:

1. Open the Edit Properties dialog for the asset that you want to edit, as described in Section 43.5.5.1, "Accessing the Edit Properties Dialog of an Asset."
2. In the **Display Name** field, edit the display name of the asset.
3. In the **Description** field, enter a description of the asset.
4. Click **OK**.

43.5.5.3 Associating an Icon with an Asset

You can associate an icon with an asset. In the current version of WebCenter Portal, the associated icon is visible only for page styles when you create a page at runtime.

To associate an icon with an asset:

1. Open the Edit Properties dialog for the asset that you want to edit, as described in Section 43.5.5.1, "Accessing the Edit Properties Dialog of an Asset."
2. In the **Icon URI** field, enter a standard URI path to the desired icon.

   For example, enter:

   ```
   /mycompany/webcenter/page/images/myimage.png
   ```

   You can specify either an absolute URL (where the URL should also work if entered in a browser address field), or a relative URL that points to an image located in the `/oracle/webcenter/siteresources/scopedMD/shared` folder of your application.
3. Click **OK**.

43.5.5.4 Categorizing an Asset

You can classify assets into relevant groups. For example, all page styles associated with Sales could be have a sales category. This value is available and exposed only in the Edit Properties dialog.

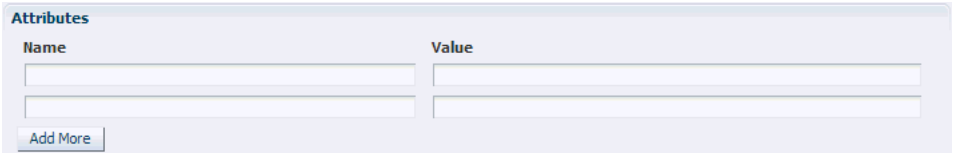
To categorize an asset:

1. Open the Edit Properties dialog for the asset that you want to edit, as described in Section 43.5.5.1, "Accessing the Edit Properties Dialog of an Asset."
2. In the **Category** field, enter a category name.
3. Click **OK**.

43.5.5.5 Setting Asset Attributes

In addition to the default attributes exposed through the Edit Properties dialog (**Display Name**, **Description**, and so on), you can expose custom attributes for assets. The Edit Properties dialog provides an **Attributes** section for entering attribute name/value pairs (Figure 43–31).

Figure 43–31 Attributes Section of an Asset



For example, `editPageAfterCreation` is a custom attribute of page styles and controls whether a newly created page of a particular style opens in Edit or View mode. It takes a value of `true` or `false`. When you associate this attribute with a particular page style, every time a user creates a page based on the selected style, the attribute value is considered and the page functions accordingly.

To set an asset attribute:

1. Open the Edit Properties dialog for the asset that you want to edit, as described in Section 43.5.5.1, "Accessing the Edit Properties Dialog of an Asset."
2. In the Attributes section, in the **Name** field, enter the attribute name.
3. In the **Value** field, enter a value for the attribute.
4. Click **Add More** if you want to add more attributes.

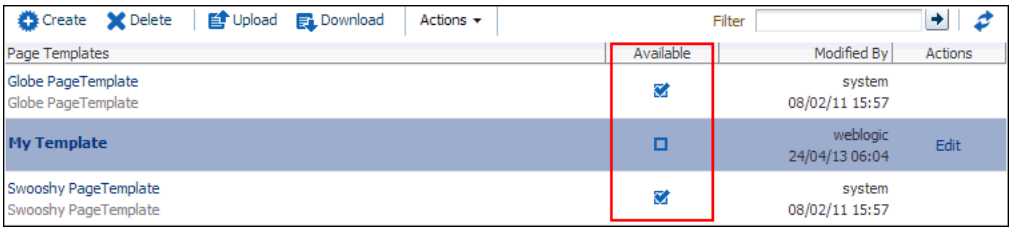
   This adds a new row to the Attributes section. You can then enter the required details in the **Name** and **Value** fields.
5. To remove an attribute that is associated with an asset, click the **Remove** icon displayed next to the attribute that you want to remove.
6. Click **OK**.

43.5.6 Showing or Hiding an Asset

When you create an asset, by default the asset is marked as hidden. A hidden asset is not available for use in the asset picker. For an asset to become available, it must be published.

For all assets that are available to use, a check box marked with a blue tick appears in the **Available** column for the asset on the Assets page. An empty check box in the **Available** column indicates that the asset is marked as hidden, as shown in Figure 43–32.

Figure 43–32 Showing or Hiding an Asset



To show or hide an asset:

1. Navigate to the **Assets** page in the administration console.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."
2. In the left navigation panel, select the desired type of asset.
3. From the list of assets, select the asset that you want to show or hide.
4. To show a hidden asset, select the empty check box in the **Available** column.

   To hide an available asset, deselect the check box in the **Available** column.

Note: To mark a page as available, select the check box in the **Show Page** column in the **Assets** page. If this check box is empty, the page is marked as hidden.

43.5.7 Setting Asset Security

You can control whether all users or only specific users or groups can access the assets in your application. By default, asset access is controlled by application-level permissions. The Security Settings dialog provides a means of setting aside application-level permissions and defining specific permissions on a selected asset.

For information about setting page access, see Section 43.5.1.3, "Setting Page Access."

To set access permissions on any asset other than a page:

1. Navigate to the **Assets** page in the administration console

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. In the left navigation panel, select the desired type of asset.
3. From the list of assets, select the asset on which you want to set access permissions.
4. From the **Actions** menu, select **Security Settings**.
5. In the Security Settings dialog, specify an access method by selecting either of the following options (Figure 43–33):

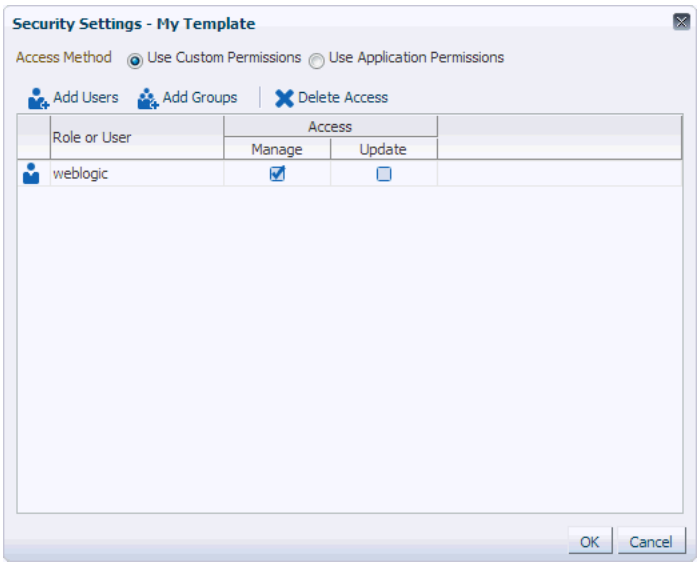
   - **Use Custom Permissions**—Select this option to define who may manage and update the selected asset. If you select this option, the other controls in the dialog become available. Proceed to step 6.

     Note: When you choose **Use Custom Permissions**, ensure that at least one user or group is granted the **Manage** access.

   - **Use Application Permissions**—Select this option to inherit the selected asset's access settings from those defined for the application. If you select this option, click **OK** to save your changes and exit the dialog.

     Note: Selecting **Use Application Permissions** removes all custom permissions that you may have set.

Figure 43–33 Security Settings Dialog for an Asset



6. If you selected **Use Custom Permissions**, specify the user or group to whom you want to grant asset access.
   - Click **Add Users** to search for and select individual users available in your identity store.
   - Click **Add Groups** to search for and select groups of users available in your identity store.
7. For each user or group selected, specify the level of access you want to grant. Select:
   - **Manage** to grant full access to the asset. Such users can perform such tasks as edit, delete, grant access, show or hide, and so on.
   - **Update** to grant the permission to edit the asset. Such users can edit the asset, but they cannot delete it.
8. If you want to revoke permissions from any user or group, select that entity and click **Delete Access**.
9. Click **OK**.

43.5.8 Uploading and Downloading an Asset

In your deployed application, you can edit assets at runtime. However, for greater control, you may want to edit an asset at design time. For this, you can download the asset created at runtime, edit it at design time in JDeveloper, and then upload the updated asset back into the application without redeploying the application.

> **Note:** Pages and pagelets cannot be downloaded or uploaded at the asset level.

The **Download** and **Upload** options enable post-deployment, round-trip application development. These actions greatly simplify the process of bringing new or revised assets from JDeveloper into your application and pushing them back into development from your application to JDeveloper as needed.

43.5.8.1 Downloading an Asset

When you download an asset, its configuration is saved into an archive file. You can save the archive file either to your local file system or a remote server file system.

To download an asset:

1. Navigate to the **Assets** page in the administration console

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. In the left navigation panel, select the desired asset type.
3. From the list of assets, select the asset you want to download.
4. In the menu bar, click **Download**.
5. In the Download dialog, in the **Archive File Name** field, enter the name of the export archive file.
6. Select:
   - **Save to My Computer** - to save the export archive file to your local file system.
   - **Save to WebCenter Portal Server** - Click this to save the export archive file to a remote server file system. In the Path field, enter the location on the server where you want to save the archive file.
7. Click **Download**.
8. After downloading the asset, you can then import it into JDeveloper.

43.5.8.2 Uploading an Asset

After editing your assets in JDeveloper, you can upload them into your application at runtime.

When you upload an asset:

- Existing assets, that is, assets with the same internal ID, are overwritten.
- Assets must be in an archive file format on your local file system or a remote server.

To upload an asset at runtime:

1. Navigate to the **Assets** page in the administration console

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. In the left navigation panel, select the desired asset type.
3. In the menu bar, click **Upload**.
4. In the Upload dialog, select:
   - **Look on My Computer** to upload an archive file from your local file system. Click **Choose File** to locate the file.
   - **Look on WebCenter Portal Server** to upload an archive file from a remote server file system. In the field below, enter the location on the server where the file is located.
5. Click **Upload**.
6. If the asset already exists in the portal, click **Yes** to confirm that you want to replace the asset with the contents of the archive file.

43.5.9 Previewing an Asset

You can edit assets at runtime, preview the changes, and make further adjustments as needed.

To preview an asset:

1. Navigate to the **Assets** page in the administration console

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. In the left navigation panel, select the desired asset type.
3. Click the name of the asset that you want to preview.

   Note: Not all assets can be previewed. If the name of the asset is not a hyperlink, then it cannot be previewed.

   The asset is displayed as it would appear when used in a portal.

4. View and interact with the asset to determine whether it looks and behaves the way you want.
5. When you are done, click **Close**.

   You can either edit the asset to make changes or publish it to make it available in the application.

43.5.10 Deleting an Asset

When an asset is no longer required, you may want to remove it from your application. You cannot delete built-in assets.

Note: Before you delete an asset, you must ensure that the asset is not in use. If you mark an asset for deletion, it is deleted even if it is use.

To delete an asset:

1. Navigate to the **Assets** page in the administration console.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. In the left navigation panel, select the desired asset type.
3. From the list of assets, select the asset you want to delete.
4. On the menu bar, click **Delete**.

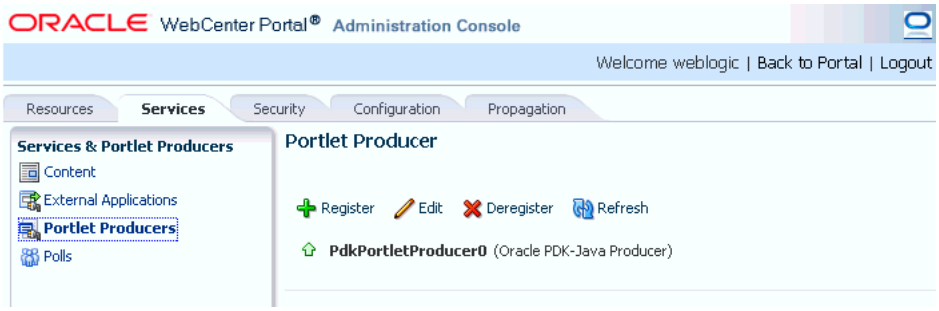
   Note: To delete a page, select the **Delete Page** option from the **Actions** menu of the page. When you delete a page, its subpages are also deleted.

5. In the Delete dialog, click **OK** to complete the process.

43.6 Configuring Services, Portlet Producers, and External Applications for Portal Framework Applications

You can manage and configure several WebCenter Portal services for a Portal Framework application through the WebCenter Portal Administration Console. From the Services tab (Figure 43–34), you can manage and configure the content repository, polls, portlet producers, and external applications.

Figure 43–34 WebCenter Portal Administration Console - Services Tab



Some services, such as Analytics, are ready to use out-of-the-box and do not require administrator-level configuration. Other services, such as Polls, require additional configuration by users with administrative privileges to get things up and running.

This section includes the following topics:

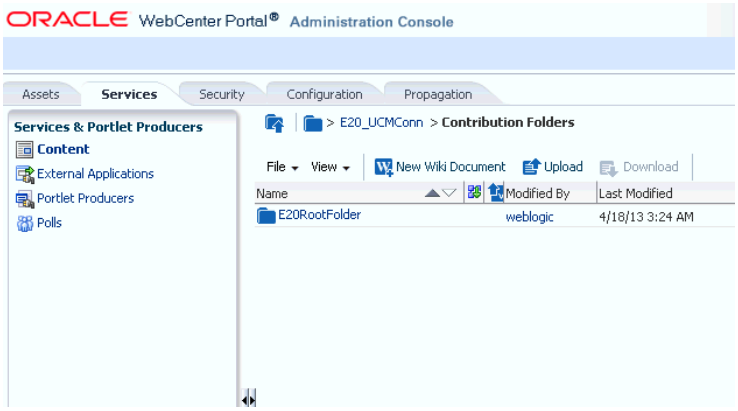
- Section 43.6.1, "Managing Content"
- Section 43.6.2, "Managing Portlet Producers"
- Section 43.6.3, "Managing External Applications"
- Section 43.6.4, "Creating and Configuring Polls"

43.6.1 Managing Content

System administrators can manage content that is stored in the application's primary content repository through the WebCenter Portal Administration Console (Figure 43–35). Administrators can add, edit and update content from here, manage document version history, and also access useful information such as direct URLs and download URLs for files and folders.

The systems administrator is responsible for registering content repositories for Portal Framework applications and determining the primary (or default) content repository. If you expected this administration page to display content from a different content repository, the primary (or default) content repository connection may need to be reconfigured. See Section 9.7, "Changing the Active (or Default) Content Repository Connection."

Figure 43–35 Contents of a Primary Documents Repository



This section includes the following topics:

- Section 43.6.1.1, "Creating a New Folder"
- Section 43.6.1.2, "Creating a Wiki Page"
- Section 43.6.1.3, "Editing a File"
- Section 43.6.1.4, "Uploading a Document"
- Section 43.6.1.5, "Checking Out a Document"
- Section 43.6.1.6, "Uploading a New Version of a Document"
- Section 43.6.1.7, "Viewing Version History of a Content Item"
- Section 43.6.1.8, "Getting Direct and Download URLs of a Document"
- Section 43.6.1.9, "Organizing Columns for the Displayed Content"
- Section 43.6.1.10, "Setting Up Security on Folders and Documents"

43.6.1.1 Creating a New Folder

To create a new folder:

1. Navigate to the Administration console and open the Services tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. Select **Content**.
3. Open a folder in which you want to create a new folder.
4. From the File menu, select **New Folder**...
5. In the Create Folder dialog, enter a descriptive name in the **Folder Name** field, and then click **Create** to create the folder.

   Tip: To rename or delete a folder, select the appropriate option from the File menu. This menu also provides options to cut, copy, and paste contents in a folder.

   To hide folders in a directory, from the View menu, select **Hide Folders**. To make hidden folders visible, deselect the **Hide Folders** option.

43.6.1.2 Creating a Wiki Page

To create a wiki document:

1. Navigate to the Administration console and open the Services tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. Select **Content**.
3. Select a folder in which you want to create your wiki page.
4. Click **New Wiki Document**.

   The Rich Text Editor displays.

5. In the **Title** field, enter a descriptive title, and click **Create** to create the wiki page in the chosen folder.

43.6.1.3 Editing a File

To edit contents such as a wiki document:

1. Navigate to the Administration console and open the Services tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

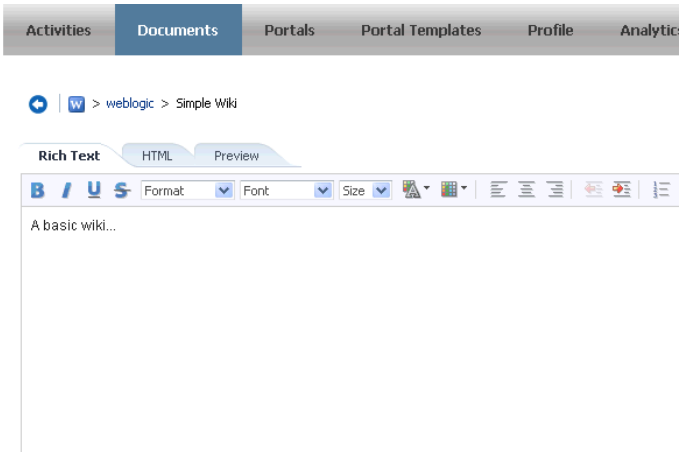
2. Select **Content**.
3. Click the item you want to edit.

   You can edit wikis, blogs, and other Rich Text-based content directly. For other file formats you must download, make your changes, and then upload the file.

4. From the File menu, select **Edit**.

   For Wikis, the file opens in the Rich Text Editor, as shown in Figure 43–36.

Figure 43–36 A Wiki Page Opened for Editing



5. Click **Save and Close** to close the document after saving.

43.6.1.4 Uploading a Document

To upload a document:

1. Navigate to the Administration console and open the Services tab.

See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. Select **Content**.
3. Select a folder into which you want to upload your document.
4. Click **Upload Document**. The Upload Document to <*Folder Name*> dialog displays.
5. In the **Upload Document** section, click **Browse** and select the required document. In the **Description** section you can enter a description if you like, and then click **Upload**. Your document is uploaded in the chosen directory.

   The maximum file upload size in Portal Framework applications is 2 MB. Portal Framework application developers can customize this limit at design time.

   Tip: You can upload multiple documents at a time. You can add more field to upload as many documents as you like by clicking **More**.

   To download a document, select it and click **Download**.

43.6.1.5 Checking Out a Document

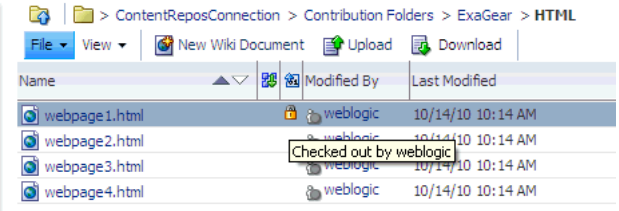
To check out a document:

1. Navigate to the Administration console and open the Services tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. Select **Content**.
3. Select the document you want to check out.
4. From the File menu, select **Check Out**. The document is checked out and the lock icon appears to indicate its checked out status, as shown in Figure 43–37.

Figure 43–37 Checked Out Document



Tip: To cancel check out, select the document in the directory it is located, and from the File menu choose **Cancel Check Out**.

43.6.1.6 Uploading a New Version of a Document

To upload a new version of a document:

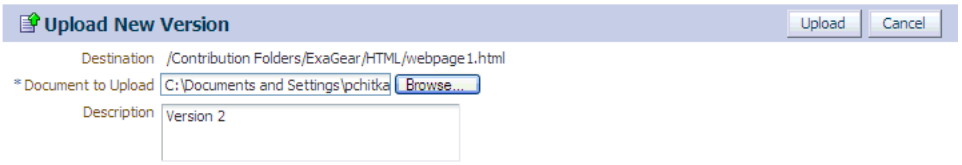
1. Navigate to the Administration console and open the Services tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. Select **Content**.
3. Select the document you want to check out.
4. From the File menu, select **Upload New Version**.

5. In the Upload New Version dialog, click **Browse** to select another version of the document, as shown in Figure 43–38. You can enter a description, if you like, and then click **Upload**.

Figure 43–38 Upload a New Version of a Document



43.6.1.7 Viewing Version History of a Content Item

To view the version history of a document such an image or a wiki page:

1. Navigate to the Administration console and open the Services tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. Select **Content**.
3. Navigate to the folder in which your documents are located.
4. Select a document to view its version history.
5. From the View menu, select **Version History**. The version history displays, as shown in Figure 43–39.

Figure 43–39 Version History of a Document

> **Tip:** To view properties of an item, select this item in the directory it is located, and from the View menu, select **Properties**.

43.6.1.8 Getting Direct and Download URLs of a Document

A direct URL lets you view a document, whereas a download URL lets you download it. To get these URLs:

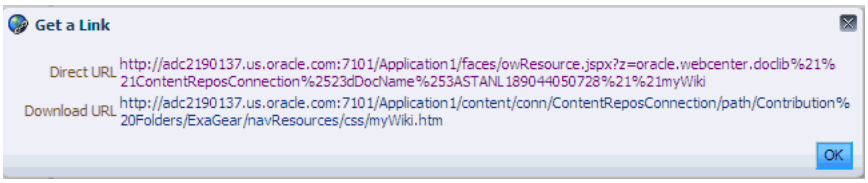
1. Navigate to the Administration console and open the Services tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. Select **Content**.
3. Navigate to the folder in which your documents are located.
4. Select a document to get its URLs.
5. From the View menu, select **Get a Link**.

6. In the Get a Link dialog, click the **Direct URL** if you want to view this document. To download this document, click **Download URL**.

Figure 43–40 Direct and Download URLs



43.6.1.9 Organizing Columns for the Displayed Content

For each item in a primary repository you can choose what associated information you would like to display, such as name of a content item, its last modified date, its check out status, and so on. You can also reorder chosen columns to display the desired information in a specific order.

43.6.1.9.1 Showing Columns

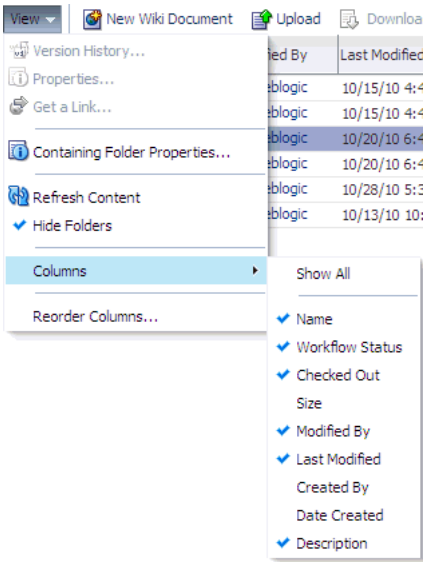
To choose columns that will display the desired information associated with your content items:

1. Navigate to the Administration console and open the Services tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. Select **Content**.
3. Navigate to the folder in which your documents are located.
4. From the View menu, select **Columns** and then choose titles that will display the desired information for your content items, as shown in Figure 43–8.

Figure 43–41 Columns - Show



43.6.1.9.2 Reordering Columns

To reorder columns:

1. Navigate to the Administration console and open the Services tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

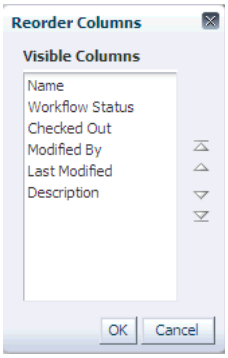
2. Select **Content**.

3. Navigate to the folder in which your documents are located.

4. From the View menu, select **Reorder Columns**.

   The Reorder Columns dialog displays, as shown in Figure 43–42.

Figure 43–42 Columns - Reorder



43.6.1.10 Setting Up Security on Folders and Documents

Under the Contents tab, in the File menu, the Security menu item for a document or folder is visible when the following conditions are met:

- Item level security has been enabled in Oracle WebCenter Content Server, as described in Section 9.2.3.10, "Configuring Item Level Security."
- If the security group assigned to these documents is listed in the `SpecialAuthGroups` setting at the time when the item level security is enabled in Oracle WebCenter Content Server, as described in Section 9.2.3.10, "Configuring Item Level Security."
- The user has administrative rights on the document or folder in Oracle WebCenter Content Server.

For information about using the security feature, see the "Setting Security Options on a Folder or File" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

43.6.2 Managing Portlet Producers

You can use the administration console to register, edit, and delete portlet producers. For more information about using the administration console to manage portlet producers, see Section 21.9, "Managing Portlet Producers with the Administration Console."

43.6.3 Managing External Applications

You can use the WebCenter Portal Administration Console to register, edit, and delete external applications. For more information about using the WebCenter Portal Administration Console to manage external applications, see Section 23.4, "Managing External Applications with the WebCenter Portal Administration Console."

43.6.4 Creating and Configuring Polls

System administrators can create and configure online polls for a Portal Framework application at runtime through the WebCenter Portal Administration Console. With polls, you can survey your audience (such as their opinions and their experience level), check whether they can recall important information, and gather feedback.

This section includes the following topics:

43.6.4.1 About Polls

With polls, in addition to taking available polls, you can do the following:

- Create a poll by clicking the **Create Poll** icon, and then adding section headings and questions to it
- Schedule the poll for distribution
- Save the poll as a template for use with new polls
- Analyze the results of the poll

Polls is integrated with many WebCenter Portal services, such as RSS, Search (to search poll text), Instant Messaging and Presence, and Recent Activities.

Figure 43–43 shows an example poll.

Figure 43–43 Example Poll

Polls must be published and open before they can be completed by users. Users cannot complete unpublished or closed polls.

43.6.4.2 Creating, Configuring, and Analyzing a Poll

From the Polls page (Figure 43–44) you can create polls, perform operations on them, including edit polls, save (as a poll or as a poll template), publish polls, close polls,

analyze the status of current polls, and delete polls. You can update poll data on the Polls page at any time by clicking the **Refresh** icon.

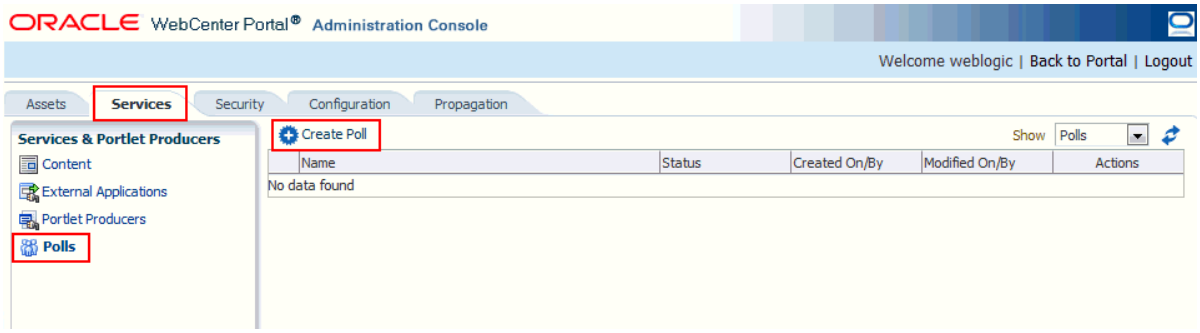
To add a new poll:

1. Click the **Services** administration tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. Click **Polls** (Figure 43–44).

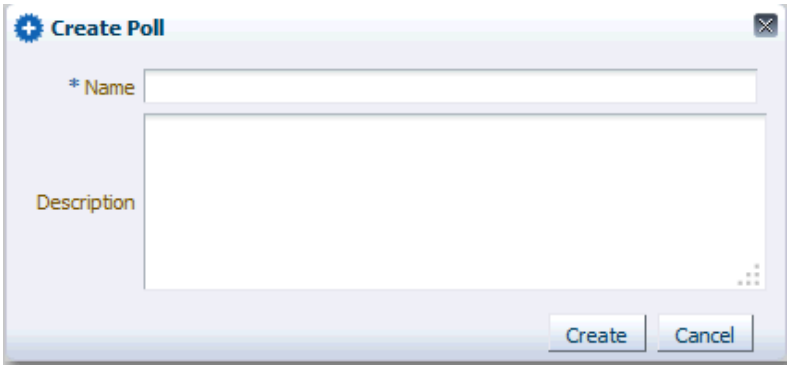
Figure 43–44 WebCenter Portal Administration Console - Polls



3. Click **Create Poll**.

   The Create Poll dialog displays (Figure 43–45).

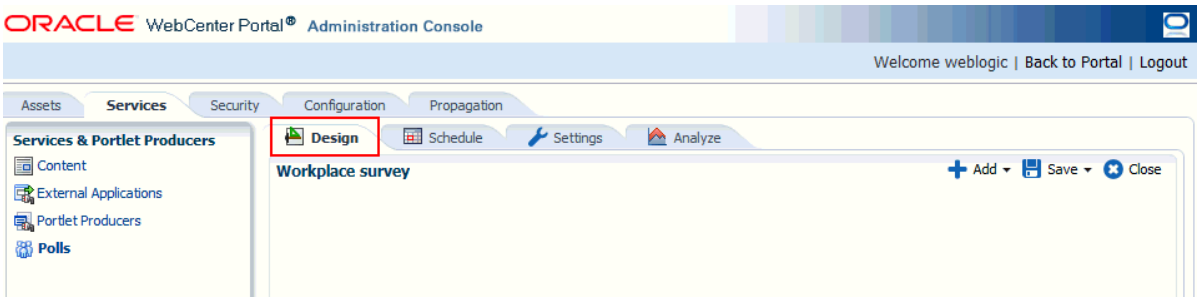
Figure 43–45 Create Poll Dialog



4. Enter a **Name** and, optionally, a **Description** for the poll, then click **Create**.

   The Design tab displays (Figure 43–46).

Figure 43–46 Create Polls - Design Tab

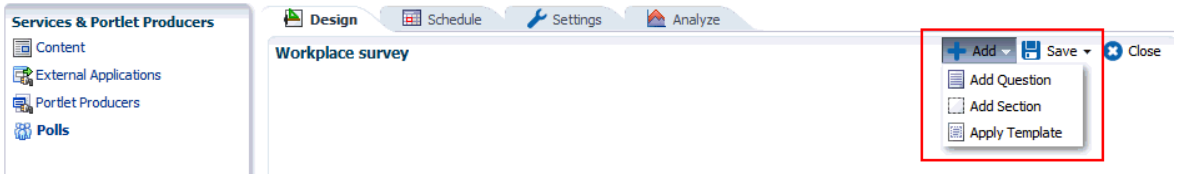


5. Click **Add** and select an option (Figure 43–47).

   Select from the following options:

   - Add Question—to add a new poll question
   - Add Section—to add a section to the poll
   - Add Template—to apply an existing template to the poll

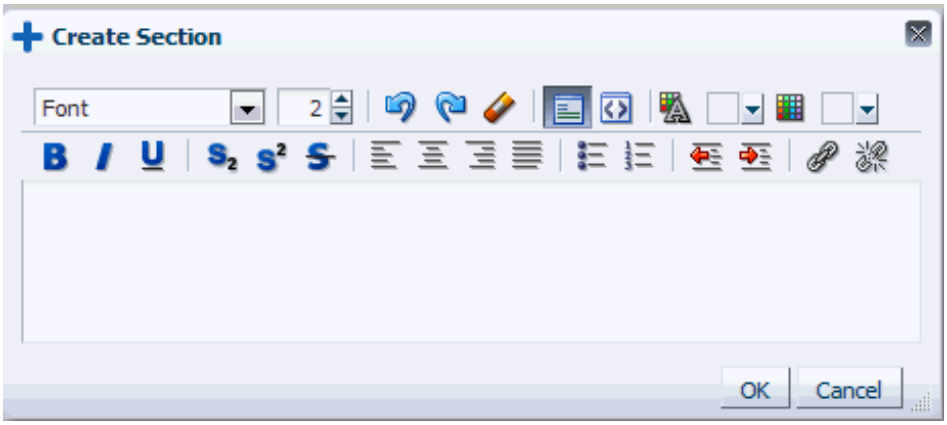
Figure 43–47 Create Polls - Add Options



6. To add a section to organize your poll questions, click **Add Section**.

   The Create Section dialog displays (Figure 43–48).

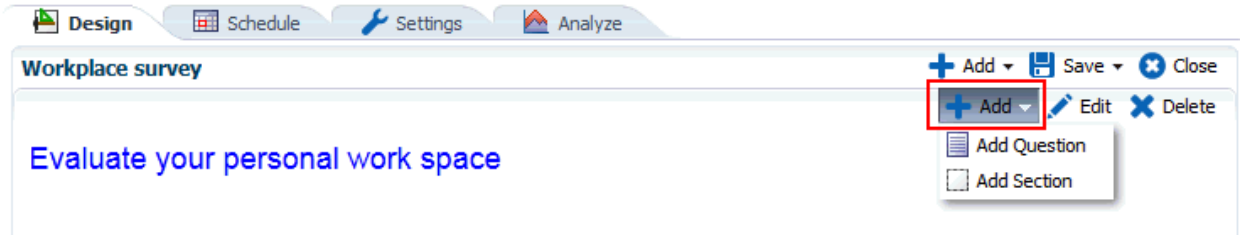
Figure 43–48 Polls - Create Section Dialog



1. Enter any explanatory text or descriptive text for the section in the rich text editor (Figure 43–48).
2. Apply formatting to the section heading, if you choose to do so.
3. Click **OK**.
4. Click **Add** and then select **Add Section** to add other sections to your poll.

7. Click **Add** in a section to add a poll question under the section and then select **Add Question** (Figure 43–49) to add each poll question.

Figure 43–49 Create Polls - Add Question

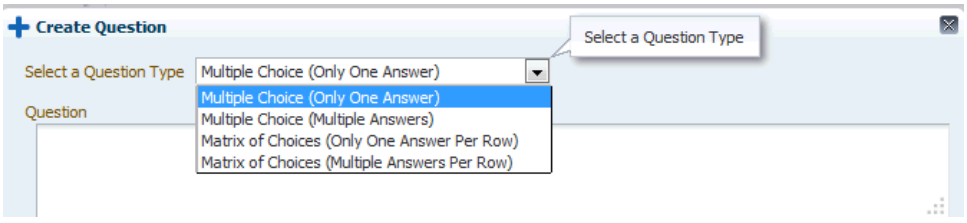


The Create Question dialog displays (Figure 43–50).

Figure 43–50 Create Question Dialog

1. Select the **Question Type** from the drop-down list (Figure 43–51):

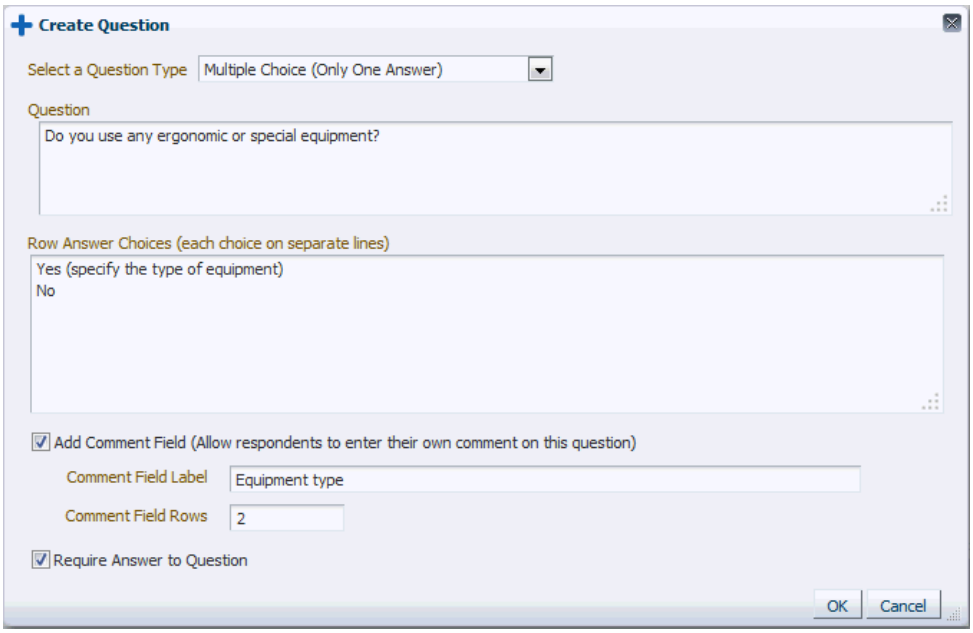
Figure 43–51 Polls - Create/Edit Question Dialog - Question Type



2. Enter the **Question** text and the **Answer Choices** (Figure 43–52).

   For multiple choice questions, each choice must be on a separate line.

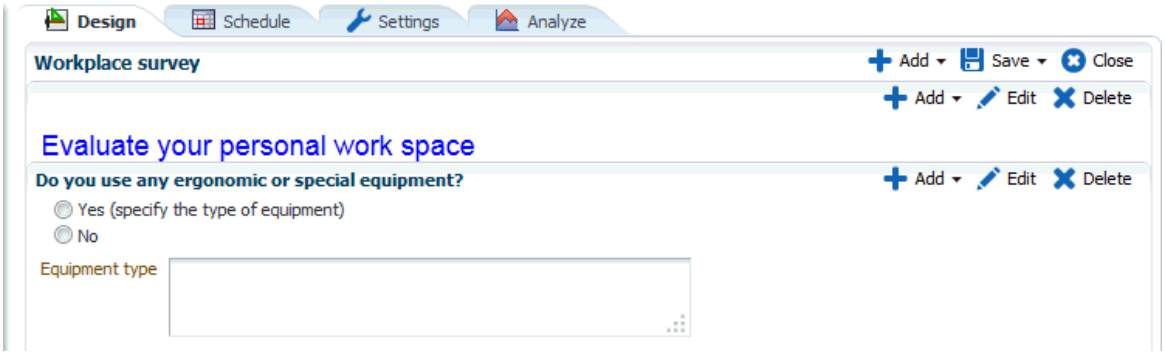
Figure 43–52 Polls - Create Question Dialog



3. Select **Add Comment Field** if you want users to add an explanation or comment to their response, then enter a label for the field and specify the number of rows.
4. Select **Require Answer to Question** if you want the question to be mandatory.
5. Click **OK**.

   For this example, the current design is shown in Figure 43–53.

Figure 43–53 Create Polls Example - Design Tab

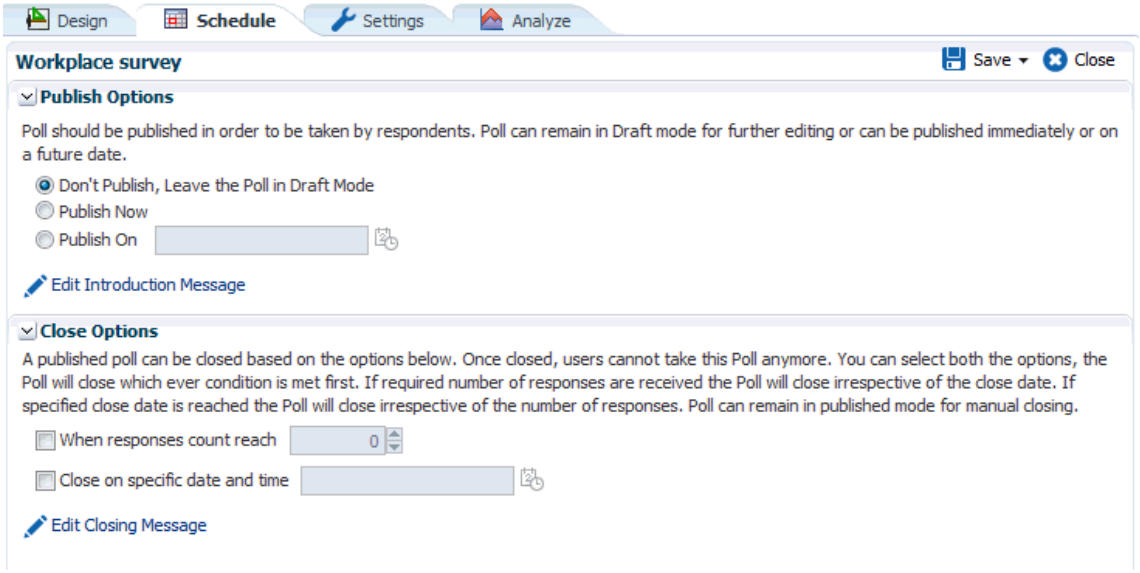


6. Click **Add** and then select **Add Question** to add other questions to the section.

8. Click the **Schedule** tab to specify publish and close options for the poll (see Figure 43–54).

   Polls must be published and open for users to take. Users cannot take unpublished or closed polls.

Figure 43–54 Create Polls - Schedule Tab



1. Make a selection from **Publish Options**.

   Select to keep the poll in draft mode for further editing, publish it immediately, or publish it on a future date.

   If you select to publish it on a future date, click the **Select Date and Time** icon to enter the publishing time through a calendar.

   Click **Edit Introduction Message** to customize the text provided at the beginning of the poll in the rich text editor.

2. Make a selection from **Close Options**.

   Select to close the published poll after it reaches a certain number of responses or on a certain date. If you select both options, then the poll closes when either condition is first met.

   If you select to close the poll on a specific date, click the **Select Date and Time** icon to enter the closing time through a calendar.

   Click **Edit Closing Message** to customize the text provided at the end of the poll in the rich text editor.

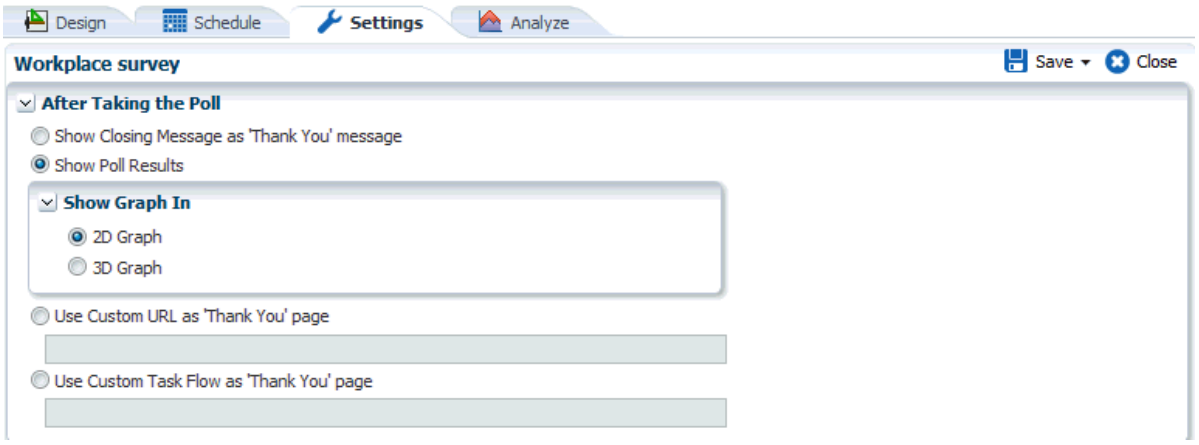
3. Click **OK**.

4. Click **Save** to save your schedule.

9. Click **Settings** to select what to display after users take the poll (Figure 43–55).

   For example, you can set a custom URL or JSF task flow as a Thank You page that appears after the poll is taken.

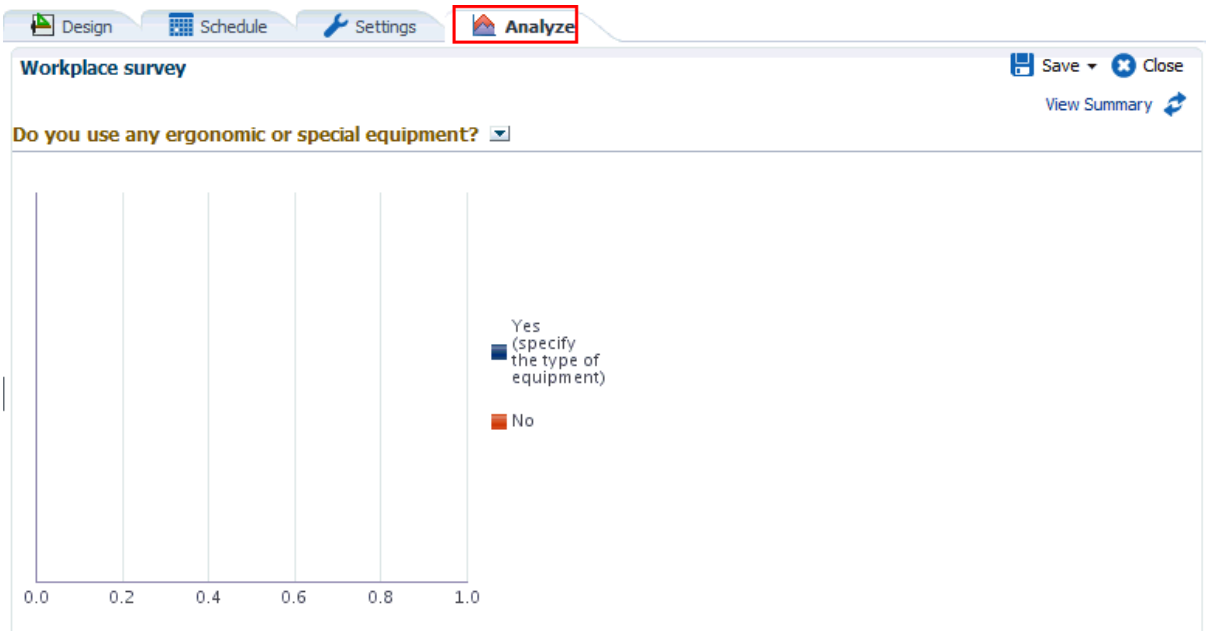
Figure 43–55 Create Polls - Settings Tab



10. After conducting the poll, open the **Analyze** tab to view the poll data. (Figure 43–56).

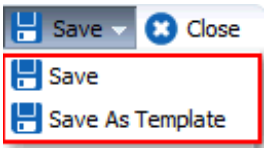
    See Section 43.6.4.6, "Analyzing the Results of a Poll."

Figure 43–56 Create Polls - Analyze Tab



11. Click **Save** to save your poll (Figure 43–57).

Figure 43–57 Saving a Poll



When you save any poll, you can select to save it as a template. After it is saved as a template, you can later apply that poll template to other polls.

Figure 43–58 shows your poll after it is published on a portal page. For users to view and take polls, add the Take Polls task flow to a page. See Section 43.6.4.7, "Taking Polls."

Figure 43–58 Poll Shown in the Take Polls Task Flow

43.6.4.3 Editing a Poll

To edit a poll:

1. Click the **Services** administration tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

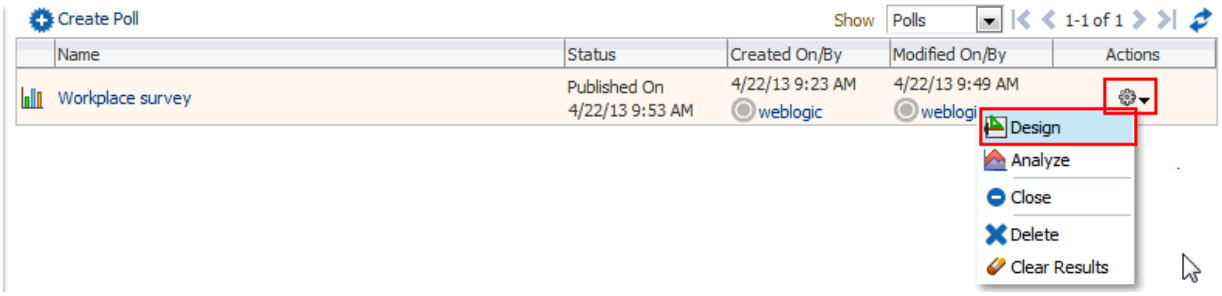
2. Click **Polls** (Figure 43–44).

3. Click the poll name link to edit the poll name and description.

4. Click the **Actions** menu and then select **Design** to make any changes (Figure 43–59).

   See Section 43.6.4.2, "Creating, Configuring, and Analyzing a Poll" for more information.

Figure 43–59 Edit Options for a Poll



5. Click **Save** and then **Close**.

43.6.4.4 Deleting a Poll

To delete a poll:

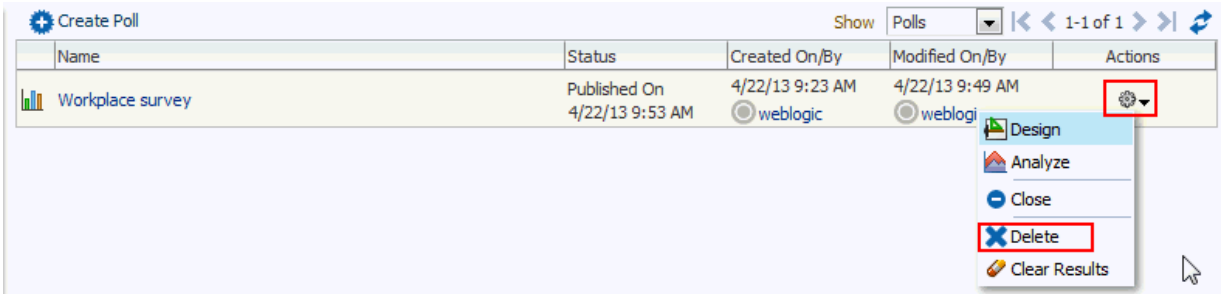
1. Click the **Services** administration tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. Click **Polls** (Figure 43–44).

3. Select the poll, click the **Actions** menu and select **Delete** (Figure 43–60).

Figure 43–60 Delete Option for a Poll



4. Click **Yes** in the Delete Poll dialog.
5. Click **Save** and then **Close**.

43.6.4.5 Closing a Poll

To close a poll:

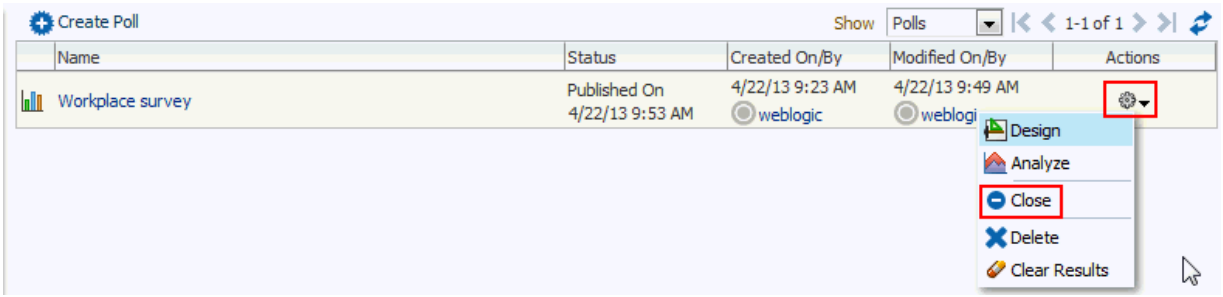
1. Click the **Services** administration tab.

   See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

2. Click **Polls** (Figure 43–44).
3. Select the poll, click the **Actions** menu and select **Close** (Figure 43–61).

   The Close option is available only after a poll is published.

Figure 43–61 Close Option for a Poll



   The Status of the poll changes to Closed, showing the date and time the poll closed.

   Tip: Click the **Actions** menu and select **Publish** if you want to open the poll again.

4. Click **Save** and then **Close**.

43.6.4.6 Analyzing the Results of a Poll

You can analyze the results for a poll after users have taken the poll.

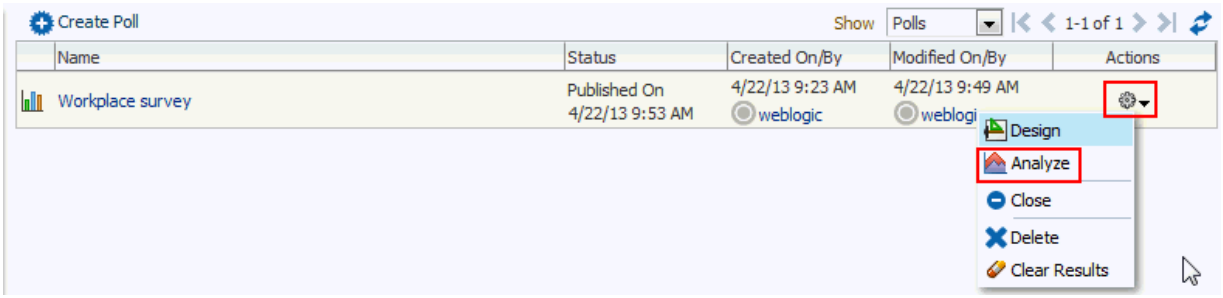
To analyze a poll:

1. Click the **Services** administration tab.

See Section 43.2, "Accessing the Administration Console for Portal Framework Applications."

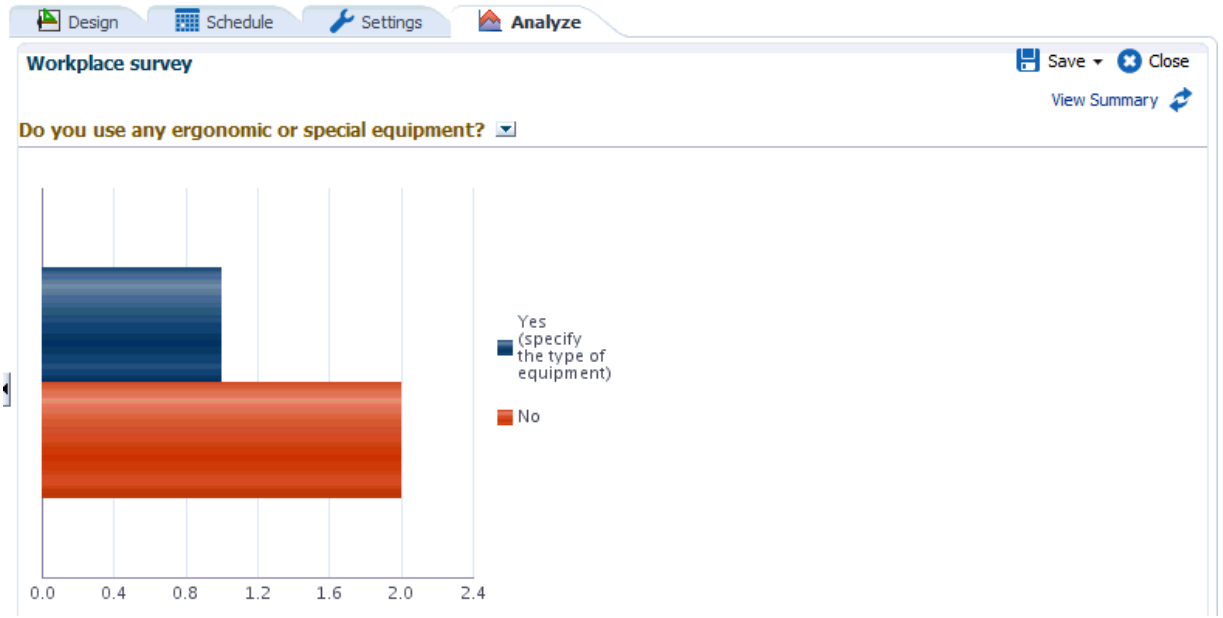
2. Click **Polls** (Figure 43–44).
3. Click the **Actions** menu and select **Analyze** (Figure 43–62).

Figure 43–62 Analyze Option for a Poll



The Analyze page appears (Figure 43–63).

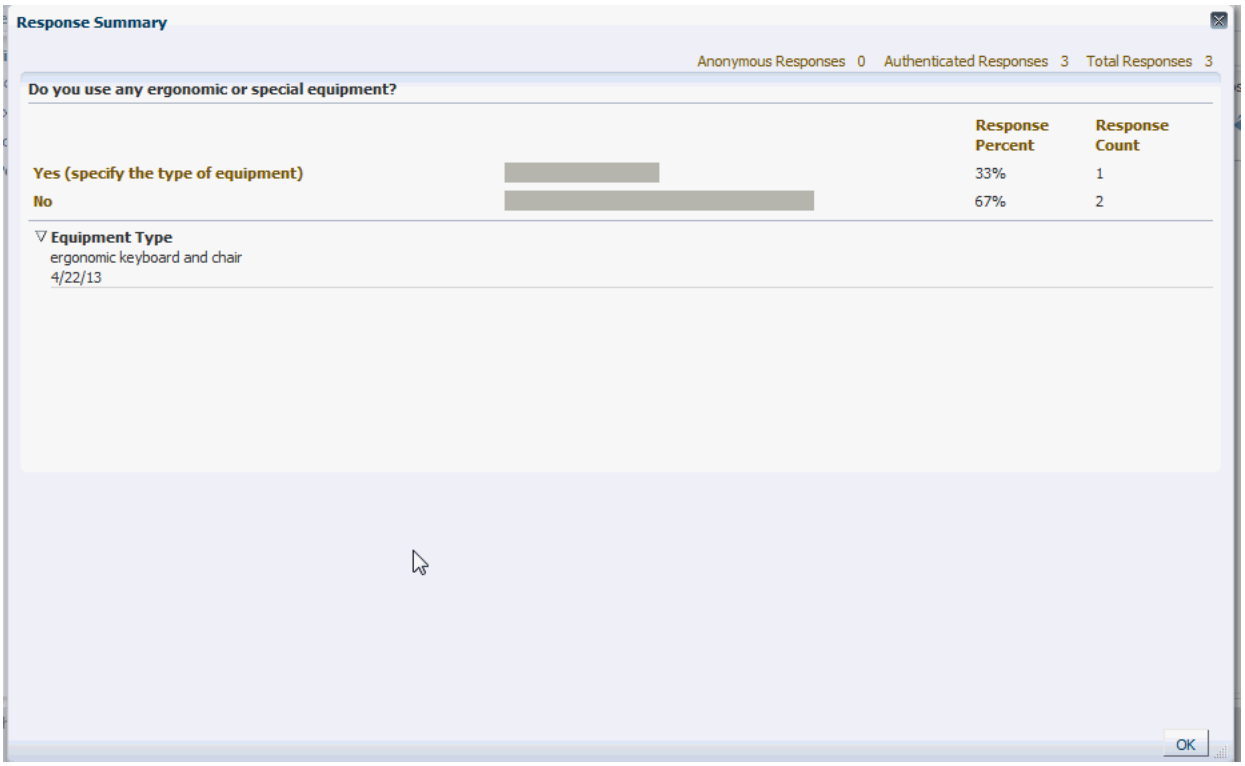
Figure 43–63 Polls Manager - Analyze Page



4. If the poll included a field for poll takers to add comments, click **View Summary** to see consolidated comments.

   The Response Summary dialog appears (Figure 43–64).

Figure 43–64 Response Summary Dialog



Expand the name of the poll's custom field name to view all comments, if you gave poll respondents an option to add comments.

43.6.4.7 Taking Polls

The Take Polls task flow (Figure 43–43) displays the most recently-published available poll, unless it is set to display a specific poll with the `Poll Id` parameter (Figure 43–65).

Note: To work with the Take Polls task flow:

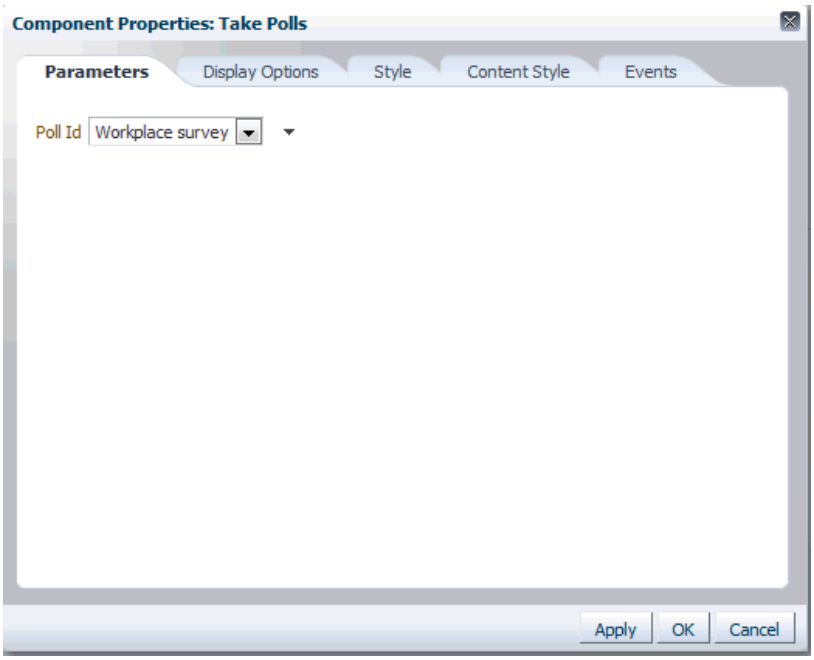
- If the Take Polls task flow is not in your resource catalog, it must be added to the resource catalog and that resource catalog must be applied to the Resource Catalog for Pages. See the "Developing Resource Catalogs" chapter in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.
- The Take Polls task flow must be added to the appropriate page in your portal. See the "Creating Pages and Adding Resources" chapter in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*

After a user submits a response for that poll, this task flow displays the next most recently-published poll.

43.6.4.8 Setting Polls Task Flow Properties

The Polls task flows have associated properties, which users with sufficient privileges can access through the Component Properties dialog in Composer (Figure 43–65).

Figure 43–65 Task Flow Component Properties



Note: See the "Polls Task Flow Parameters" table in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper* for information.

43.7 Propagating Portal Framework Application Changes From Staging to Production

The WebCenter Portal Administration Console includes a propagation tool for moving Portal Framework application metadata from a staging to a production server. Site administrators use this tool occasionally to push approved changes to the production server without incurring any downtime. For more details, see the "Using the Propagation Tool to Propagate From Staging to Production" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

44

Managing Export, Import, Backup, and Recovery for Portal Framework Applications

This chapter describes the export, import, backup, and recovery capabilities and tools available for Portal Framework applications.

This chapter includes the following topics:

- Section 44.1, "Exporting and Importing Portal Framework Applications for Data Migration"
- Section 44.2, "Backing Up and Recovering Portal Framework Applications"

Portal Framework applications store data related to configuration and content for the various features in several locations. To facilitate disaster recovery and the full production lifecycle from development through staging and production, Oracle WebCenter Portal provides a set of utilities that enable you to back up this data, and move the data between Portal Framework applications in staging and production environments.

If you want to migrate a test instance to a production instance, see the "Moving Oracle WebCenter Portal to a Target Environment" section in *Oracle Fusion Middleware Administrator's Guide* which describes an alternative migration approach.

Permissions: To perform the tasks in this chapter, you must be granted the WebLogic Server `Admin` role through the Oracle WebLogic Server Administration Console.

See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

44.1 Exporting and Importing Portal Framework Applications for Data Migration

This section describes how to export and import metadata and application customizations for Portal Framework applications developed using Oracle JDeveloper.

It includes the following sections:

- Section 44.1.1, "Understanding Portal Framework Application Export and Import"
- Section 44.1.2, "Prerequisites for Portal Framework Application Export and Import"
- Section 44.1.3, "Exporting Portlet Client Metadata for Portal Framework Applications"

- Section 44.1.4, "Importing Portlet Client Metadata for Portal Framework Applications"
- Section 44.1.5, "Exporting Portal Resources for Portal Framework Applications"
- Section 44.1.6, "Importing Portal Resources for Portal Framework Applications"
- Section 44.1.7, "Exporting Metadata for Portal Framework Applications"
- Section 44.1.8, "Importing Metadata for Portal Framework Applications"
- Section 44.1.9, "Migrating Security for Portal Framework Applications"
- Section 44.1.10, "Migrating Schema Data for Portal Framework Applications"

44.1.1 Understanding Portal Framework Application Export and Import

Several migration tools are available to export and import Portal Framework application, their connections and customizations (that is, customizations applied to an application, pages, and portlets) between stage and production environments (Figure 44–1).

Figure 44–1 Portal Framework Application Export and Import

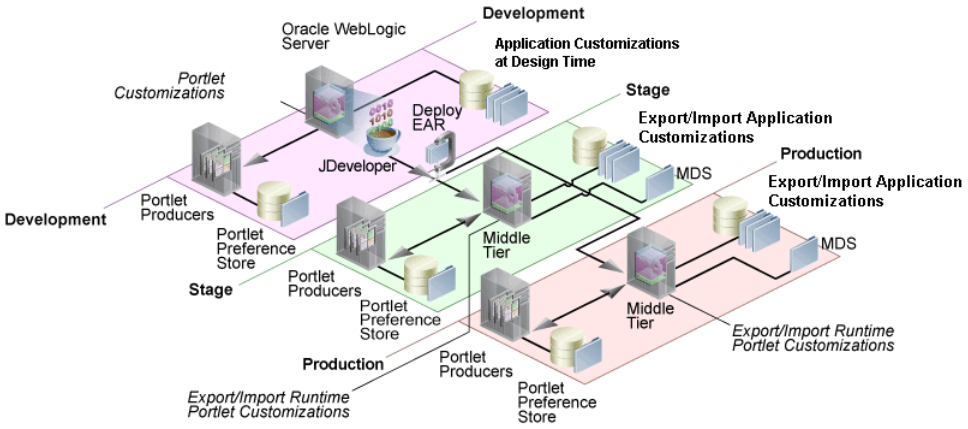


Table 44–1 lists available migration tools and their capabilities. All customizations listed in Table 44–1 are migrated with Portal Framework applications.

Table 44–1 Portal Framework Application Migration Tools

Migration Tools	Capabilities
Portlet Client WLST Commands	Enable export and import of portlet client metadata, and portlet customizations and personalizations.
Portal Resource WLST Commands	Enable export and import of portal resources, such as skins, page templates, and so on.
MDS WLST Commands	Enables export and import of: ■ Portal Framework application metadata including customizations made to pages, tools, and services ■ Data stored in the `connections.xml` and `adf-config.xml` documents
Security Migration WLST Commands	Enables export and import of security policies, including roles and mapping of users and roles.

Table 44–1 (Cont.) Portal Framework Application Migration Tools

Migration Tools	Capabilities
Oracle Database Utilities	Enables export and import of Portal Framework application data. For information, see the "Oracle Data Pump" part in *Oracle Database Utilities* guide.
Non-Oracle database utilities	Refer to the database manufacturer's documentation for information about their data migration tools.

44.1.2 Prerequisites for Portal Framework Application Export and Import

Before exporting or importing metadata and customizations for a Portal Framework application, ensure the following:

- The database in which the application metadata and schema is stored is up and running.
- The target instance is configured with the same set of tools and services as the source instance. Additional tools and services can be configured in the target, if required, but minimally, the configuration in the source and target must match.
- The `jps.policystore.removal` parameter is set to `OFF` in your application's `weblogic-application.xml` so that policies are migrated on import:

```
<application-param>
    <param-name>jps.policystore.removal</param-name>
    <param-value>OFF</param-value>
  </application-param>
```

  If this option is not set, no policy information is imported. In some instances you may not want to migrate policy data, for example, when migrating from a test environment to a production environment where test data is not required. Note however, that pages created on the source instance at runtime do not display on the target instance because no page grants exist on the target.

44.1.3 Exporting Portlet Client Metadata for Portal Framework Applications

To export portlet client metadata and producer customizations and personalizations for a Portal Framework application, use the WLST command `exportPortletClientMetadata`. This command is run on the entire application, and therefore, it exports metadata of all the producers stored in an application. You cannot opt to export metadata for specific producers.

Before you run the command, ensure that the custom managed server on which the Portal Framework application is deployed, the portlet producers, and the database in which the application metadata or schema is stored are up and running.

For detailed syntax and examples, see the "exportPortletClientMetadata" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

For information on how to export portlet client metadata associated with all applications, see the "How to Manage the Persistence Store for JSR 286 Portlets" and "Migrating a PDK-Java Producer Persistence Store" sections in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

44.1.4 Importing Portlet Client Metadata for Portal Framework Applications

To import portlet client metadata and producer customizations and personalizations for a Portal Framework application, use the WLST command `importPortletClientMetadata`.

Prerequisites:

- The database in which the application metadata or schema is stored and the portlet producers must be up and running.
- Use the WLST command `exportPortletClientMetadata` to export the portlet client metadata and producer customizations and personalizations to an EAR file, See also, Section 44.1.3, "Exporting Portlet Client Metadata for Portal Framework Applications".

For detailed syntax and examples, see the "importPortletClientMetadata" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*. See also, "Metadata Services (MDS) Custom WLST Commands".

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

44.1.5 Exporting Portal Resources for Portal Framework Applications

Authorized users can download portal resources, such as skins and page templates, while a Portal Framework application is running, edit and extend them in tools such as Oracle JDeveloper, and then upload them back to the Portal Framework application. Users who want to share or migrate portal resources to other Portal Framework applications can use the download feature too.

You can download the following portal resources at runtime through the Portal Framework's Administration Console:

- Skins
- Page styles
- Page templates
- Content Presenter display templates
- Navigations
- Resource catalogs
- Task flows
- Task flow styles

When you download (or export) a portal resource, the resource details are saved to an export archive (.EAR). You can save the export archive to your local file system or a remote server file system using a filename of your choice.

For details, see Section 43.5.8, "Uploading and Downloading an Asset".

Alternatively, system administrators can perform the same task using the WLST command `exportWebCenterResource`. For command syntax and examples, see the "exportWebCenterResource" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

44.1.6 Importing Portal Resources for Portal Framework Applications

Authorized users can import portal resources, such as skins and page templates, while a Portal Framework application is running. You can import the following portal resources at runtime through the WebCenter Portal Administration Console:

- Skins
- Page styles
- Page templates
- Content Presenter templates
- Navigations
- Resource catalogs
- Task flows
- Task flow styles

You can import portal resources previously saved to WebCenter Portal export archive files (.ear), on your local or remote server file system. Existing portal resources, that is, resources with the same internal ID are overwritten on import.

For details, see Section 43.5.8, "Uploading and Downloading an Asset".

Alternatively, administrators can perform the same task using the WLST command `importWebCenterResource`. For command syntax and examples, see the "importWebCenterResource" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

44.1.7 Exporting Metadata for Portal Framework Applications

Portal Framework application metadata associated with pages, task flows, tools, and services is stored in the Oracle metadata store (MDS). For detailed information about MDS, see the "Managing the MDS Repository" chapter in *Oracle Fusion Middleware Administrator's Guide*.

For a particular Portal Framework application, you can use the `exportMetadata` command to export all base documents and their customizations stored in MDS or metadata for a specific tool or service.

For example:

```
exportMetadata(application='myWebCenterApp', server='WC_CustomPortal',
toLocation='/tmp/myrepos', docs='/oracle/webcenter/**')
```

Where:

- `application`: Name of the Portal Framework application for which the metadata is to be exported (for example, `myWebCenterApp`).
- `server`: Server on which the Portal Framework application is deployed (for example, `WC_CustomPortal`).
- `toLocation`: Target directory to which documents selected from the source partition are to be exported. The `toLocation` parameter can be used as a temporary file system for migrating metadata from one server to another.

- `docs`: List of comma separated fully qualified document name(s) and/or document name patterns (* and ** patterns).

  In this example, "`docs='/oracle/webcenter/**`" exports the required documents for *all* Oracle WebCenter Portal's tools and services that store metadata in MDS.

  Note: The "`docs='/oracle/webcenter/**`" command *does not* export portlet customizations and personalizations or changes to configuration files such as `connections.xml` and `adf-config.xml`.

  - To export portlet metadata, run the WLST command `exportPortletClientMetadata`. See also, Section 44.1.3, "Exporting Portlet Client Metadata for Portal Framework Applications".
  - To export configuration file updates that are stored in MDS, run the WLST command `exportMetadata` with "`docs='/META-INF/mdssys/cust/adfshare/adfshare/**`". See also, Appendix A.1.1, "adf-config.xml and connections.xml".

For detailed syntax and examples, see the "exportMetadata" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Metadata, which consists of base and customization documents, are stored in the following paths:

- **Analytics**:`/oracle/webcenter/analytics/**`
- **Announcements**: `/oracle/webcenter/collab/announcement/**`
- **Blogs**: `/oracle/webcenter/blog/**`
- **Documents**: `/oracle/webcenter/doclib/**` and `/oracle/webcenter/doclib/view/jsf/fragments/**`
- **Discussions**: `/oracle/webcenter/collab/forum/**`
- **General Settings**: `/oracle/webcenter/generalsettings/**`
- **Events**:`/oracle/webcenter/collab/events/**`
- **External Applications**:`/oracle/webcenter/admin/**` and `oracle/adfinternal/extapp/**`
- **Instant Messaging and Presence**: `/oracle/webcenter/collab/rtc/**`
- **Links**: `/oracle/webcenter/relationship/**`
- **Language**: `/oracle/webcenter/webcenterapp/**`
- **Lists**: `/oracle/webcenter/list/**` and `/oracle/webcenter/list/view/jsf/regions/**`
- **Mail**: `/oracle/webcenter/collab/mail/**`
- **Navigations**: `/oracle/webcenter/navigationtaskflows/**`
- **Notes**: `/oracle/webcenter/note/**`

- **Page**: `/oracle/webcenter/page/** and /pageDefs/**`
- **Polls**: `/oracle/webcenter/collab/survey/**`
- **People Connections (Connections)**:`/oracle/webcenter/peopleconnections/connection/**`
- **People Connections (Feedback)**:`/oracle/webcenter/peopleconnections/kudos/**`
- **People Connections (Profile Gallery)**:`/oracle/webcenter/peopleconnections/personalweb/**`
- **People Connections (Profile)**:`/oracle/webcenter/peopleconnections/profile/**`
- **People Connections (Message Board)**:`/oracle/webcenter/peopleconnections/wall/**`
- **Polls**: `/oracle/webcenter/collab/survey/**`
- **Recent Activity**: `/oracle/webcenter/recentactivity/**`
- **Resource Action Handler**: `/oracle/webcenter/framework/service/**`
- **RSS News Feed**: `oracle/webcenter/rssviewer/**`
- **Scope**: `/oracle/webcenter/framework/scope/**`
- **Search**: `/oracle/webcenter/search/**`
- **Security**: `/oracle/webcenter/security/**`
- **Smart Tag**: `/oracle/webcenter/collab/smarttag/**`
- **Portal Browser**: `/oracle/webcenter/community/**`
- **Portal Contacts**: `/oracle/webcenter/people/**`
- **Subscriptions**: `/oracle/webcenter/notification/**`
- **Tags**: `/oracle/webcenter/tagging/**`
- **adf-config.xml, connections.xml**: `/META-INF/mdssys/cust/adfshare/adfshare/**`

  Configuration file updates are not stored under the `/oracle/webcenter/` directory alongside Oracle WebCenter Portal's tools and services metadata. To export customizations associated with these files, run `exportMetadata` again with `"docs='META-INF/mdssys/cust/adfshare/adfshare/**"`. See also, Appendix A.1.1, "adf-config.xml and connections.xml".

44.1.8 Importing Metadata for Portal Framework Applications

To import tools and services metadata and customizations for a Portal Framework application, use the WLST command `importMetadata`. For example:

```
importMetadata(application='myWebCenterApp', server='WC_CustomPortal',
fromLocation='/tmp/myrepos', docs='/**')
```

Where:

- `application`: Name of the Portal Framework application for which the metadata is to be imported (for example, `myWebCenterApp`).
- `server`: Name of the target server on which the application is deployed (for example, `WC_CustomPortal`).

- `fromLocation`: Source directory from which documents are imported. The `fromLocation` parameter can be any temporary file system location for migrating metadata from one server to another.
- `docs`: List of comma separated fully qualified document name(s) and/or document name patterns (* and ** patterns).

For detailed syntax and examples, see the "importMetadata" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

See also, Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

44.1.9 Migrating Security for Portal Framework Applications

Security migration involves moving the identity store, credential store, and policy store, from one Portal Framework application to another. The migration process for Portal Framework applications is exactly the same as that described for the out-of-the-box application "WebCenter Portal". For details, see:

- Section 41.6.6, "Backing Up and Restoring LDAP Identity Store"
- Section 41.6.7, "Backing Up and Restoring Policy Stores (LDAP and Database)"
- Section 41.6.8, "Backing Up and Restoring Credential Stores (LDAP and Database)"

44.1.10 Migrating Schema Data for Portal Framework Applications

Use export and import database utilities to migrate or back up application schema data for a Portal Framework application. This section includes the following subsections:

- Section 44.1.10.1, "Understanding Schemas Used by Portal Framework Applications"
- Section 44.1.10.2, "Exporting Schema Data for Portal Framework Applications"
- Section 44.1.10.3, "Importing Schema Data for Portal Framework Applications"

44.1.10.1 Understanding Schemas Used by Portal Framework Applications

Oracle WebCenter Portal components require that various schemas are installed in a supported database. Your Portal Framework applications may use one or more of the following schemas:

- `WEBCENTER` - For using tags, links, lists, polls, and people connections in your application
- `PORTLET` - For storing portlet producer data and customizations
- `ACTIVITIES` - For storing activity graph and analytics data
- `DISCUSSIONS` and `DISCUSSIONS_CRAWLER` - For storing discussions and announcements
- `OCS` - For storing documents in Oracle WebCenter Content Server

44.1.10.2 Exporting Schema Data for Portal Framework Applications

To export data associated with a Portal Framework application, use the appropriate database utility:

- For an Oracle database, go to *`ORACLE_HOME`*`/bin` of your database and run the `expdp` command.

For example, see Example 44–1 which shows how to export the WEBCENTER schema. The example shows that before exporting the schema data, you must create a directory where the exported data will be stored. You can use similar command for other schemas listed in Section 44.1.10.1, "Understanding Schemas Used by Portal Framework Applications."

- For non-Oracle databases, refer to the manufacturer's documentation.

Example 44–1 Oracle Data Pump Utility (Exporting WEBCENTER Schema)

```
sqlplus "sys/password@serviceid as sysdba"
create or replace directory dmpdir as 'full_path_to_directory_on_file_system';
GRANT read,write ON directory dmpdir TO public;
exit;

DB_ORACLE_HOME/bin/expdp \"sys/password@serviceid as sysdba\"
directory=dmpdir dumpfile=wcp.dmp SCHEMAS=srcprefix_WEBCENTER EXCLUDE=STATISTICS
NOLOGFILE=Y
```

Where:

- *DB_ORACLE_HOME* is the directory in which the database for the WebCenter Portal schema (WEBCENTER) is installed.
- *password* is the password for system database user. For example, mydb1234.
- serviceid is the service ID of the database connection.
- directory is the location on the database machine where the dump file will be created.
- dumpfile identifies the name of the file that will contain the exported data.
- SCHEMAS identifies the schema to be exported. Include the RCU suffix that was used during installation (_WEBCENTER), along with a user supplied prefix. For example, DEV_WEBCENTER.
- EXCLUDE=STATISTICS specifies not to export statistics for the tables.
- NOLOGFILE=Y Suppresses the creation of a log file.

For detailed expdp command information, see the *Oracle Database Utilities* guide.

44.1.10.3 Importing Schema Data for Portal Framework Applications

To import Portal Framework application data, use the appropriate database utility:

- For an Oracle database, go to *ORACLE_HOME*/bin of your database and run the impdp command.

  For example, see Example 44–2 which shows how to import the WEBCENTER schema. You can use similar command for other schemas listed in Section 44.1.10.1, "Understanding Schemas Used by Portal Framework Applications."

- For non-Oracle databases, refer to the manufacturer's documentation.

Example 44–2 Oracle Data Pump Utility (Importing WEBCENTER Schema)

```
DB_ORACLE_HOME/bin/sqlplus "sys/password@serviceid as sysdba"
create or replace directory dmpdir as 'mydmpdirectory';
GRANT read,write ON directory dmpdir TO public;
drop user tgtprefix_WEBCENTER cascade;
exit;
```

```
DB_ORACLE_HOME/bin/impdp  \"sys/password@serviceid as sysdba\"
directory=dmpdir dumpfile=wcp.dmp SCHEMAS=tgtprefix_WEBCENTER
```

Where:

- *DB_ORACLE_HOME* is the directory in which the database for the WebCenter Portal schema (WEBCENTER) is installed.
- *password* is the password for system database user.
- serviceid is the service ID of the database connection.
- directory is the location on the database machine where the dump file is located.
- dumpfile is the name of the file that contains data to be imported.
- SCHEMAS identifies the target schema to be imported. Schema names include the RCU suffix that was used during installation (_WEBCENTER), along with a user supplied prefix. For example, DEV_WEBCENTER.

  Use the SCHEMAS parameter when schema names on the source and target match. For example, both schemas are named DEV_WEBCENTER.

  If schema names do not match, use the REMAP_SCHEMA parameter instead. For details, see Example 41–3, "Importing WebCenter Portal Schema Data (Source and Target Schema Names Different)".

For detailed impdp command information, see the *Oracle Database Utilities* guide.

44.2 Backing Up and Recovering Portal Framework Applications

To recover data from disasters, such as the loss of database hardware, inadvertent removal of data from file or database, it is important to back up Portal Framework applications on a frequent basis. The frequency of backup depends on how often the underlying information stored for the application changes in a particular customer application, and how much time and amount of information could acceptably be lost. Incremental or partial backups may be applied where the data is critical to the business and must be restored due to a failure.

Backup and recovery of Portal Framework applications and WebCenter Portal components can be managed through database export and import utilities, and various other tools. For more information, see the "Advanced Administration: Backup and Recovery" part in *Oracle Fusion Middleware Administrator's Guide*.

Part X

Multilanguage Portals

This part of *Oracle Fusion Middleware Administering Oracle WebCenter Portal* presents language and translation topics for Oracle WebCenter Portal.

Part X contains the following chapter:

- Chapter 45, "Managing a Multilanguage Portal"

45

Managing a Multilanguage Portal

This chapter describes the language support available in WebCenter Portal and how to manage translations at the application and portal level and for specific strings in a portal.

This chapter includes the following topics:

- Section 45.1, "About Languages in WebCenter Portal"
- Section 45.2, "Modifying and Translating Strings at the Application Level"
- Section 45.3, "Translating Strings for a Portal"
- Section 45.4, "Modifying and Adding Translations for a Specific String of a Portal"
- Section 45.5, "Adding Support for a New Language to WebCenter Portal"

Permissions: To perform the tasks in this chapter, you must be granted the following roles:

- **WebLogic Server:** `Admin` or `Monitor` role granted through the Oracle WebLogic Server Administration Console.
- **WebCenter Portal**: `Administrator` role granted through Portal Builder Administration or a custom role that grants the following permission:

  `Basic Services: Edit Page Access, Structure, and Content permission`.

See also, Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

45.1 About Languages in WebCenter Portal

If your portal must support different languages, you can configure it to display localized content based on the user's selected language and locale. For example, if you know your page will be viewed by users who speak Italian, you can localize your page so that when Italian is selected (in browser, user preferences, portal, or application settings), text strings in the page appear in Italian.

Additionally, locale selection applies special formatting considerations that are applicable to the selected locale. For example, those considerations may include whether information is typically viewed from left to right or right to left, how numbers are depicted (such as monetary information), and so on.

There are three main types of information that are displayed in WebCenter Portal:

- User interface (UI) elements, like field and button labels and seeded boilerplate text
- User-entered metadata, including page names, the portal name, and the portal description
- Content added by users, such as announcements, documents, and discussion forum content

Each type of information is handled differently when it comes to modification:

- UI elements:

  Note: UI elements include out-of-the-box translations for 28 languages and 100 different locales. You need to change this text only if the default UI text is not suited to your company's needs or if your company must support additional languages.

  – To change the text for your entire WebCenter Portal application (rather than just one portal), edit the strings in the override bundle, `SpacesSeedDataOverrideBundle`.xlf.

  – To change the UI text for a particular portal, edit the strings in the portal-specific resource bundle, `scope-resource-bundle.xlf`.

- User-entered metadata (such as page names, the portal name, and the portal description) is saved as strings in the resource bundle for the portal. Each portal has its own resource bundle. To change the user-entered metadata, edit the strings in the portal-specific resource bundle.

  Note: Generally, the user-entered metadata you want to display in multiple languages is company-wide content or customer-facing content that likely has translations available in some form. More specific content (for example, content specific to a particular department or region) is probably necessary in only one language, and therefore does not require translation.

- Content added by users is generally displayed in the language used by the contributing user.

45.1.1 Languages Supported Out-of-the-Box by WebCenter Portal

WebCenter Portal provides runtime translations for 28 languages and 100 different locales.

The list in Table 45–1 includes all the languages available to WebCenter Portal out-of-the-box. Users can also select locales associated with particular languages. For example, a user can change the language to Arabic and, within that language group, select from 20 different locales, including Algeria, Bahrain, Djibouti, and so on.

Table 45–1 Languages Available for WebCenter Portal

A to Ge	Gr to Ro	Ru to T
Arabic	Greek	Russian
Brazilian Portuguese	Hebrew	Simplified Chinese

Table 45–1 (Cont.) Languages Available for WebCenter Portal

A to Ge	Gr to Ro	Ru to T
Czech	Hungarian	Slovak
Danish	Italian	Spanish
Dutch	Japanese	Swedish
English	Korean	Thai
Finnish	Norwegian	Traditional Chinese
French	Polish	Turkish
French-Canada	Portuguese	
German	Romanian	

Note: Administrative tier that offers services to WebCenter Portal, including Oracle Enterprise Manager, provides a subset of the languages available to WebCenter Portal. These include:

- English
- Brazilian Portuguese
- Simplified Chinese
- Traditional Chinese
- French
- German
- Italian
- Japanese
- Korean
- Spanish

Discussions use WebCenter Portal's discussions server. Out-of-the-box, the discussions server application supports English and Spanish. It does not support other languages listed in Table 45–1. However, the application is open to your own translation files. For more information, refer to the Jive documentation site. This information is explicit to the discussion server application user interface.

Note: The Pagelet Producer Administration UI supports 9 administration languages and Dutch.

45.2 Modifying and Translating Strings at the Application Level

Whether you are modifying or translating UI text application-wide, UI text for a particular portal, or user-entered metadata in a portal, the process is basically the same: you just modify different files.

To modify seeded UI text application-wide, you edit the override bundle, `SpacesSeedDataOverrideBundle.xlf`.

To modify or translate strings at the application level:

1. Start WLST. For information, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."
2. Use the WLST command `exportMetadata` to export the override bundle:

   ```
   exportMetadata(application='webcenter',server='WC_Spaces',toLocation='/tmp/meta
   data',docs='/xliffBundles/SpacesSeedDataOverrideBundle.xlf')
   ```

   This example exports `SpacesSeedDataOverrideBundle.xlf` for WebCenter Portal (`webcenter`) on the `WC_Spaces` managed server to the `/tmp/metadata` folder. Always use `webcenter` as the application name.

   Change the `server` value to match the name of the managed server that hosts your installation of WebCenter Portal.

   Change the `toLocation` value to the location into which you want to export the string files.

   For more information, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands." See also "exportMetadata" in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.
3. Navigate to the folder into which you exported `/xliffBundles/SpacesSeedDataOverrideBundle.xlf`,

   Caution: Make sure to correctly encode your edited file or you receive an error when you try to import the translations. Oracle recommends using Oracle JDeveloper to edit the file because it automatically encodes special characters correctly.
4. If you want to modify the strings in the base language, open `/xliffBundles/SpacesSeedDataOverrideBundle.xlf` in a text editor.

   If you want to translate the file into another language, create a language-specific version of the file. For example, to translate the application-wide UI text into Catalina, name the file `SpacesSeedDataOverrideBundle_ca.xlf`. For translation, you will generally need to send this file to the translation team, which will update the file and send it back to you.
5. Find the `<trans-unit>` blocks you want to modify or translate.

   The `ID` attribute in `SpacesSeedDataOverrideBundle.xlf` corresponds to the resource key of the UI element displayed in Composer in WebCenter Portal.

   For example, here is the `<trans-unit>` block for the Announcements title in application-wide `SpacesSeedDataOverrideBundle.xlf` file.

   ```
   <trans-unit id="ANNOUNCEMENTS.TITLE">
   <source>Announcements</source>
   </trans-unit>
   ```
6. Edit the text in the `<source>` block to fit your business needs, then save the file.
7. Use the WLST command `importMetadata` to import the updated string file back into WebCenter Portal. For example:

   ```
   importMetadata(application='webcenter',server='WC_Spaces',fromLocation='/tmp/me
   tadata',docs='/xliffBundles/SpacesSeedDataOverrideBundle.xlf')
   ```

This example imports the string file from the `/tmp/metadata` folder to the `webcenter` application on the `WC_Spaces` managed server. Change the `fromLocation` path to the location from which you want to import the string files. Always use "`webcenter`" as the application name. Change server name to match the server that hosts your installation of WebCenter Portal.

For details, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands." See also "importMetadata" in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

8. Restart the `WC_Spaces` managed server, and confirm that the changes you made appear in the UI.

45.3 Translating Strings for a Portal

To translate strings of a particular portal, you edit the portal-specific resource bundle, `scope-resource-bundle.xlf`. The strings that can be translated are portal display name, description, and page titles.

To translate strings for a portal:

1. Start WLST. For information, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."
2. Use the WLST command `exportMetadata` to export the string files:
   - To export all string files, do not include the `docs` attribute. For example:

     ```
     exportMetadata(application='webcenter',server='WC_Spaces',toLocation='/tmp/metadata')
     ```

     This example exports all string files for WebCenter Portal (`webcenter`) on the `WC_Spaces` managed server to the `/tmp/metadata` folder. Always use `webcenter` as the application name.

     Change the value for `server` to match the name of the managed server that hosts your installation of WebCenter Portal.

     Change the `toLocation` path to the location into which you want to export the string files.

   - To export only specific string files, include the `docs` attribute. For example:

     ```
     exportMetadata(application='webcenter',server='WC_Spaces',toLocation='/tmp/metadata',docs='/oracle/webcenter/translations/scopedMD/PORTAL_GUID/scope-resource-bundle.xlf')
     ```

     This example produces similar results to the first example, but exports only a portal-specific resource bundle. Replace *PORTAL_GUID* with the GUID of the portal for which you are modifying strings.

     Note: To export more than one file, separate file locations with commas.

     For more information, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands." See also "exportMetadata" in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

3. Navigate to the folder into which you exported the string files.

Caution: Make sure to correctly encode your edited file or you receive an error when you try to import the translations. Oracle recommends using Oracle JDeveloper to edit the file because it automatically encodes special characters correctly.

4. If you want to modify the strings in the base language, open `/oracle/webcenter/translations/scopedMD/PORTAL_GUID/scope-resource-bundle.xlf`, replacing *PORTAL_GUID* with the GUID of the portal for which you are modifying strings.

   If you want to translate the file into another language, create a language-specific version of the file, and open it in a text editor. For example, to translate the portal UI text into Catalina, name the file `scope-resource-bundle_ca.xlf`.

5. Find the `<trans-unit>` blocks you want to translate.

   The `OBJECTGUID` attribute in `scope-resource-bundle.xlf` corresponds to the resource key of the UI element displayed in Composer in WebCenter Portal.

   For example, following is the `<trans-unit>` block for the display name of a page in a portal-specific `scope-resource-bundle.xlf` file:

   ```
   <trans-unit
   id="SCOPEGUID:s2f80d470_6cc4_479a_884c_9feb574b35d6:Pagedf7eed1_13eea02290b__7f
   f6:SERVICEID:oracle.webcenter.page:OBJECTTYPE:page:OBJECTGUID::PAGES.:Page2.jsp
   x.DISPLAY_NAME">
   <source>Personal25</source>
   </trans-unit>
   ```

6. Edit the text in the `<source>` block to fit your business needs, then save the file.

7. Use the WLST command `importMetadata` to import the updated string files back into WebCenter Portal. For example:

   - To import all string files, do not include the `docs` attribute. For example:

     ```
     importMetadata(application='webcenter',server='WC_Spaces',fromLocation='/tm
     p/metadata')
     ```

     This example imports all string files from the `/tmp/metadata` folder to the `webcenter` application on the `WC_Spaces` managed server. Change the `fromLocation` path to the location from which you want to import the string files. Always use "`webcenter`" as the application name. Change server name to match the server that hosts your installation of WebCenter Portal.

   - To import only specific string files, include the `docs` attribute:

     ```
     importMetadata(application='webcenter',server='WC_Spaces',fromLocation='/tm
     p/metadata',docs='/oracle/webcenter/translations/scopedMD/PORTAL_GUID/scope
     -resource-bundle.xlf')
     ```

     This example produces similar results to the first example, but imports only a portal-specific resource bundle. Replace *PORTAL_GUID* with the GUID of the portal for which you are modifying strings. It is recommended that you use the `docs` attribute.

     Note: To import more than one file, separate file locations with commas.

For details, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands." See also "importMetadata" in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

8. Restart the `WC_Spaces` managed server, and confirm that the changes you made appear in the UI.

45.4 Modifying and Adding Translations for a Specific String of a Portal

To suit your business needs, you may want to translate only a specific string of a portal. For example, you may want to translate only the title of the Announcements task flow in a specific instance on a page in a portal.

To add translation for a specific instance of a string in a portal:

1. Use the WLST command `exportMetadata` to export the string file. For example:

```
exportMetadata(application='webcenter',server='WC_Spaces',toLocation='/tmp/metadata',docs='/oracle/webcenter/translations/scopedMD/PORTAL_GUID/scope-resource-bundle.xlf')
```

   This example exports a portal-specific resource bundle for WebCenter Portal (`webcenter`) deployed on `WC_Spaces` to the `/tmp/metadata` folder. Replace *`PORTAL_GUID`* with the GUID of the portal for which you are modifying strings. If necessary, change the server name to match your WebCenter Portal installation. You must change the `toLocation` path to the location into which you want to export the string files.

   For more information, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands." See also "exportMetadata" in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

2. Create a language/locale-specific version of the string file you want to translate.

   Copy `/oracle/webcenter/translations/scopedMD/`*PORTAL_GUID*`/scope-resource-bundle.xlf`, replacing *PORTAL_GUID* with the GUID of the portal. Then save the file with the required name. For example, to translate the portal-wide UI text into Catalina, name the file `scope-resource-bundle_ca.xlf`.

3. Send the files to be translated to your translation team to edit. Translation will involve the following steps:

   a. Open the string file in JDeveloper or a text editor.

   Caution: Make sure to correctly encode your edited file or you receive an error when you try to import the translations. Oracle recommends using Oracle JDeveloper to edit the file because it automatically encodes special characters correctly.

   b. Find the `<trans-unit>` blocks you want to modify.

   Here is an example of a `<trans-unit>` block from a portal-specific `scope-resource-bundle.xlf` file. The `SCOPEGUID` shows the internal ID of the selected portal, and the `OBJECTGUID` shows the ID of the Announcements task flow.

```
<trans-unit id="SCOPEGUID:s7735bad2_2e7d_4d73_a360_423a64bfc111:SERVICEID:oracle.webcenter.peopleconn:OBJECTTYPE:profile:OBJECTGUID:ANNOUNCEMENTS.TITLE">
```

```
<source>Announcements</source>
</trans-unit>
```

The `OBJECTGUID` attribute (in `scope-resource-bundle.xlf`) corresponds to the resource key displayed for the required string in Composer. For information about resource key, see the "Finding the Resource Key for a String" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

c. Translate the `<source>` text in the specified `<trans-unit>` block as required.

d. Save the file.

4. Use the WLST command `importMetadata` to import the updated string file back into WebCenter Portal. For example:

```
importMetadata(application='webcenter',server='WC_Spaces',fromLocation='/tmp/me
tadata',docs='/oracle/webcenter/translations/scopedMD/PORTAL_GUID/scope-resourc
e-bundle.xlf')
```

This example imports the string file of the specified portal from the `/tmp/metadata` folder to WebCenter Portal (`webcenter`) deployed on the `WC_Spaces` managed server. Replace *`PORTAL_GUID`* with the GUID of the portal for which you are modifying strings. If necessary, change the managed server name to match your WebCenter Portal installation. Change the `fromLocation` path to the location from which you want to import the string files.

For details, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands." See also "importMetadata" in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

5. Update the resource key of the desired UI element in WebCenter Portal. For example, if you want to translate the title of a specific Announcements task flow in a portal, perform the following steps:

a. Log on to WebCenter Portal, go to the desired portal, and open the Component Properties dialog for the Announcements task flow whose title you want to translate. For information about accessing the Component Properties dialog, see the "Modifying Components" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

b. On the **Display Options** tab, in the **Text** field, specify a new resource key in the following format:

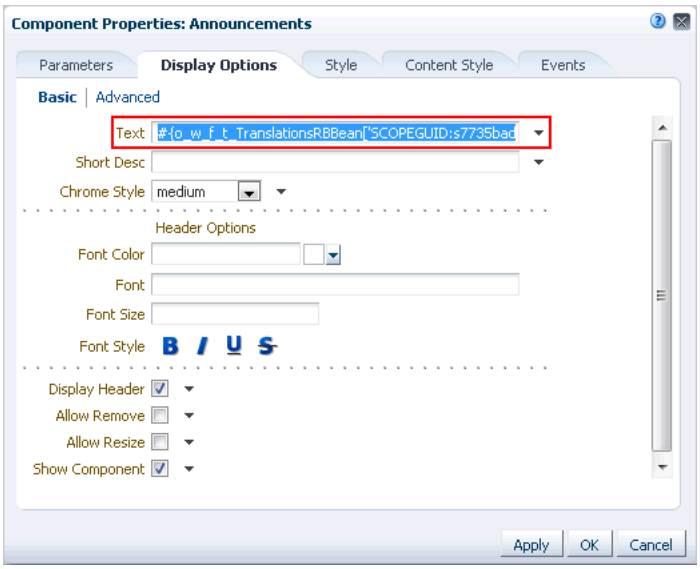
```
#{o_w_f_t_TranslationsRBBean['trans-unit id']}
```

Where, *`trans-unit id`* refers to the ID of the Announcements task flow in the `scope-resource-bundle.xlf` file. For example, specify the following resource key:

```
#{o_w_f_t_TranslationsRBBean['SCOPEGUID:s7735bad2_2e7d_4d73_a360_423a64bfc1
11:SERVICEID:oracle.webcenter.peopleconn:OBJECTTYPE:profile:OBJECTGUID:ANNO
UNCEMENTS.TITLE']}
```

c. Click **OK** (Figure 45–1).

Figure 45–1 Component Properties Dialog of the Announcements Task Flow



6. Restart the `WC_Spaces` managed server, and verify that your changes appear in WebCenter Portal.

45.5 Adding Support for a New Language to WebCenter Portal

You can add support for a new language that is not supported out-of-the-box in WebCenter Portal. To enable WebCenter Portal to support an additional language, you must translate portal strings into the new language within a resource bundle, update two language configuration files (`supported-languages.xml` and `faces-config.xml`), and then deploy your language updates to a custom shared library.

For information about adding support for a new language, see the "Using Spaces Extension Samples" white paper on the Oracle WebCenter Portal White Papers and Technical Notes page on Oracle Technology Network.

Part XI

Managing Portals in Portal Builder Administration

This part of *Oracle Fusion Middleware Administering Oracle WebCenter Portal* presents Portal Builder administration topics.

Part XI contains the following chapters:

46

Exploring the Portals Page in Portal Builder

This chapter describes how to manage portals on the **Portals** page in Portal Builder.

This chapter includes the following topics:

Permissions: To perform the tasks in this chapter on any portal, you must have the WebCenter Portal `Administrator` role or a custom role that grants the following permission:

- `Portal Server-Manage Configuration`

If you are a portal moderator (or have Manage Configuration or `Manage Configuration` permissions in a portal), you can perform the tasks in this chapter on that portal when you are in Portal Builder.

For more information about permissions, see Section 49.3, "About Application Roles and Permissions."

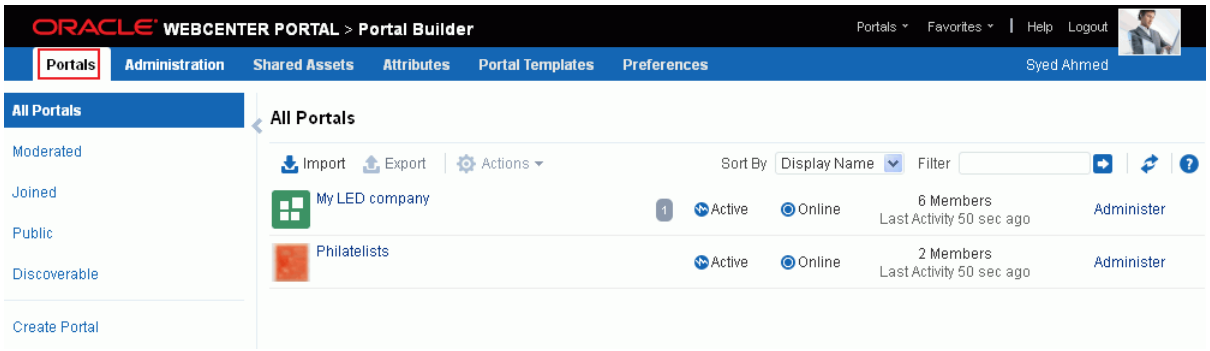
46.1 About the Portals Page in Portal Builder

The **Portals** page in Portal Builder (Figure 46–1) provides access to editing and administering all portals in WebCenter Portal.

The **Portals** page also offers import and export services that enable you to back up or move portals to stage or production environments. Portal export and import is available only to system administrators or users with application-level administrative permissions. To export and import *portal templates*, you do not need application-level administrative permissions (see the "Exporting and Importing Portal Templates" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*).

Other users will encounter this **Portals** page by either selecting **Portal Builder** from the **Portals** menu, or clicking **Back to All Portals** when administering a single portal. If granted appropriate permissions, users can use this page edit or administer portals. However, this chapter is addressed to a system administrator, who can perform administrative actions on all portals. Managing individuals portals that you create or have permissions to manage is covered in the "Administering a Portal" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Figure 46–1 WebCenter Portal Administration - Portals Page in Portal Builder

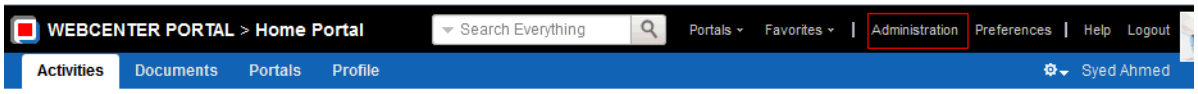


46.2 Accessing the Portals Page in Portal Builder

To manage all portals in WebCenter Portal:

1. Open the Portals page in Portals Builder in either of the following ways:
   - From the **Portals** menu, select **Portal Builder**.
   - Click the **Administration** link in the top menu bar (Figure 46–2), then click the **Portals** tab.

Figure 46–2 WebCenter Portal Administration Link



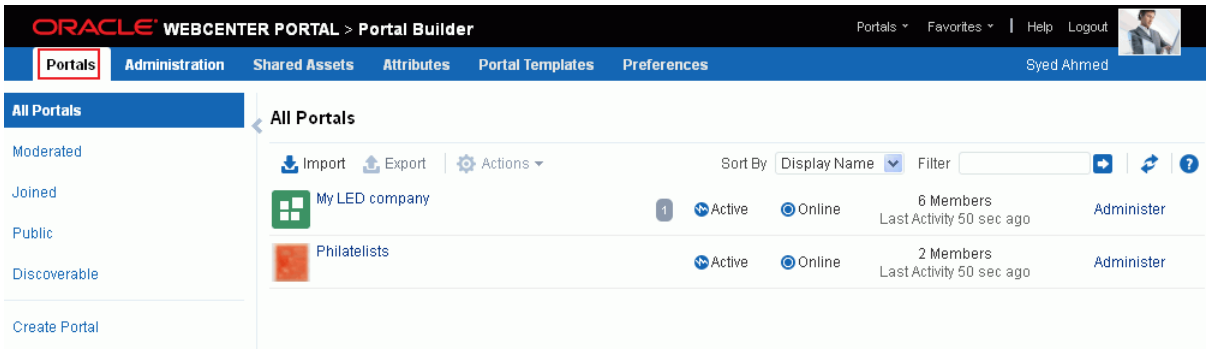
   - Enter the following URL in your browser to navigate directly to the **Portals** page:

     `http://host:port/webcenter/portal/builder/portals`

See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

The **Portals** page displays (Figure 46–3).

Figure 46–3 WebCenter Portal Administration - Portals Page in Portal Builder



2. On the **Portals** page in Portal Builder, in the left pane, select:
   - **All Portals** to show all portals that are available to you, both public and private. Portals defined as *hidden* when created are not shown.
   - **Moderated** to display portals for which you have moderator privileges.
   - **Joined** to display portals of which you are a member.
   - **Public** to display portals accessible by anyone with the portal URL.
   - **Discoverable** to display portals that can be found in search results.

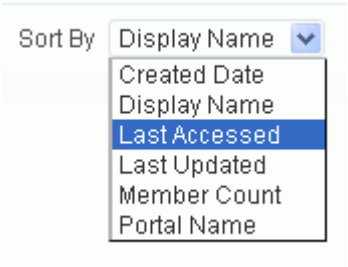
46.3 Sorting the Portals Listing

To sort the list of portals on the **Portals** page:

1. On the **Portals** page in Portal Builder (see Section 46.2, "Accessing the Portals Page in Portal Builder"), click the **Sort By** selection list (Figure 46–4).

   Note: When **All Portals** is selected in the left selection pane, you can sort by only **Display Name** and **Last Accessed**.

Figure 46–4 Sorting the Portals Listing



2. Choose a display order for the portals on the page:
   - **Created Date** to order from most to least recently created.

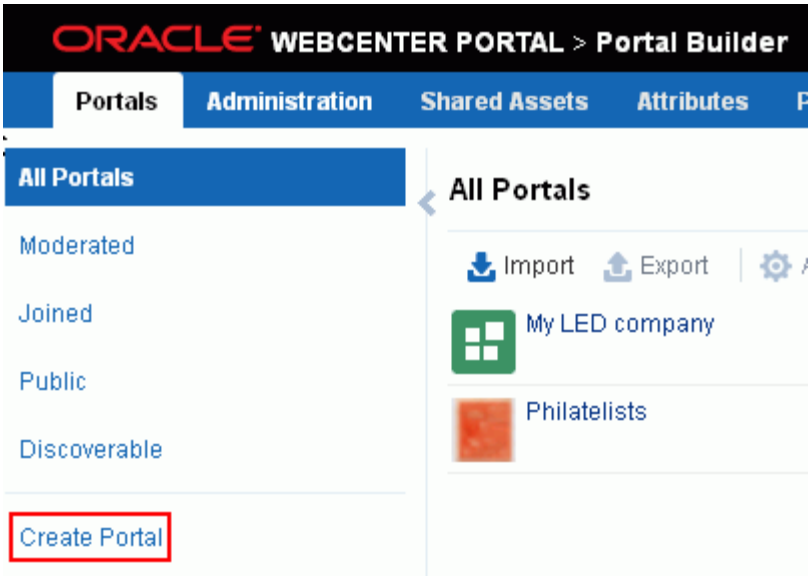
- **Display Name** to order alphabetically by external display name, as specified by its Title value in the portal administration.
- **Last Accessed** to order from most to least recently viewed, whether or not it was updated.
- **Last Updated** to order from most to least recently updated.
- **Member Count** to order by greatest to least number of portal members.
- **Portal Name** to order alphabetically by internal name of the portal, as specified by its Name value in the portal administration. The internal name is not visible on the **Portals** page.

46.4 Creating a Portal

To create a new portal:

1. On the **Portals** page in Portal Builder (see Section 46.2, "Accessing the Portals Page in Portal Builder"), click **Create Portal** (Figure 46–5).

Figure 46–5 Creating a New Portal



The **Select a Portal Template** page appears.

For information about creating a portal, see the "Creating and Building a New Portal" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

46.5 Importing or Exporting a Portal

Only system administrators can import and export portals. For more information, see:

- Section 40.1.2.4.2, "Importing a Portal from an Archive Using Portal Builder Administration"
- Section 40.1.2.3.2, "Exporting Online Portals to an Archive Using Portal Builder Administration"

See also Section G.8, "Troubleshooting Individual Portal and Portal Template Import and Export."

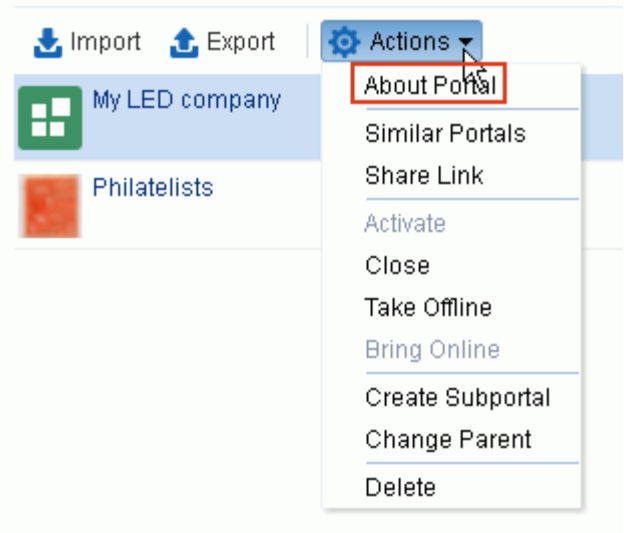
46.6 Viewing Information About Any Portal

On the **Portals** page in Portal Builder, you can quickly see whether portals are active, online, offline, how recently a portal was accessed, and membership counts.

To view information about a portal:

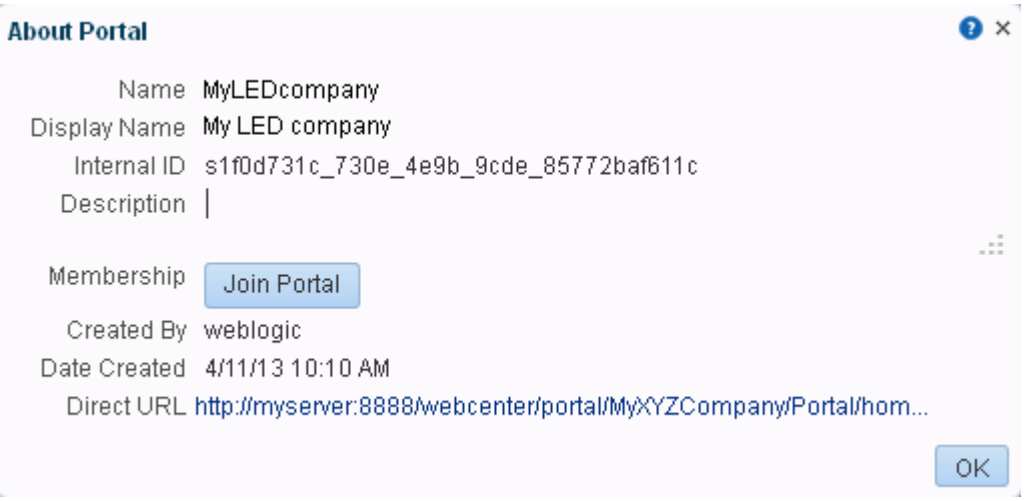
1. On the **Portals** page in Portal Builder (see Section 46.2, "Accessing the Portals Page in Portal Builder"), select the required portal by highlighting the row in the table.
2. From the **Actions** menu, select **About Portal** (Figure 46–6).

Figure 46–6 Viewing Information About a Portal



3. Explore the information in the About Portal dialog (Figure 46–7):

Figure 46–7 About Portal Dialog



- **Name**: Internal name of the portal displayed in the portal URL.

- **Display Name**: Display name of the portal. This name displays at the top of the portal and other places where portals are available for selection, such as the **Portals** page.
- **Internal ID**: ID of the portal, which other applications may use to reference this portal.
- **Description**: A description of the portal, specified when creating the portal or in the portal administration settings.
- **Created By**: User name of the portal creator.
- **Date Created**: Date and time that the portal was created.
- **Direct URL**: URL that provides direct access to the portal.

46.7 Viewing Similar Portals

A portal is considered similar to another portal if the same people perform similar actions in it, especially if they edit the content. Similar Portals is provided by the Activity Graph service. For more information, see .

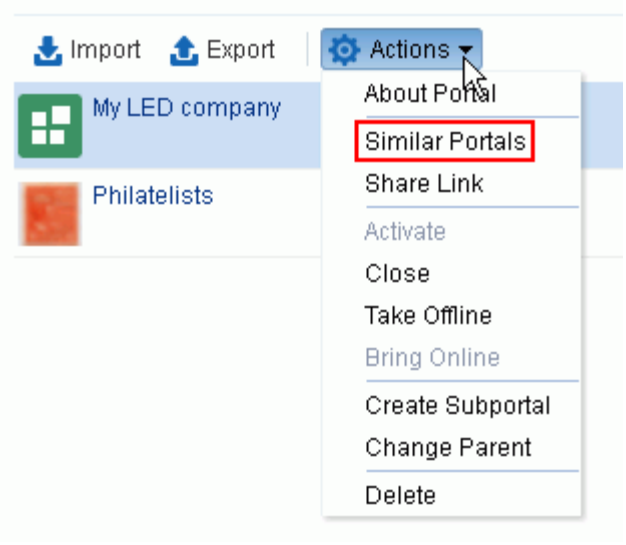
To view a list of portals similar to the current portal:

1. On the **Portals** page in Portal Builder (see Section 46.2, "Accessing the Portals Page in Portal Builder"), select the required portal by highlighting the row in the table.

   Press Ctrl+click to select more than one portal.

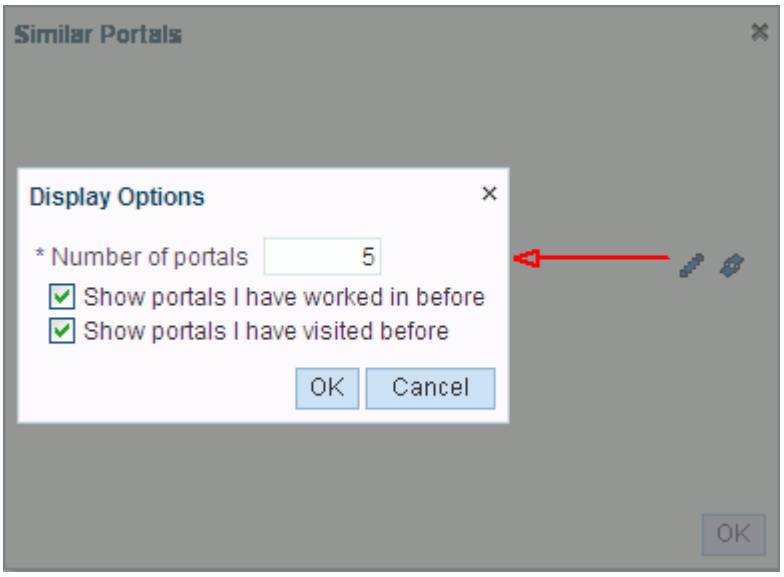
2. From the **Actions** menu, select **Similar Portals** (Figure 46–8).

Figure 46–8 Viewing Similar Portals



3. In the Similar Portals dialog, select a portal to open, or click the **Display Options** (pencil) icon (Figure 46–9) to modify the criteria for similar portals.

Figure 46–9 Similar Portals Display Options Dialog



4. In the Display Options dialog (Figure 46–9), enter the number of portals to display and the display criteria, then click **OK**.

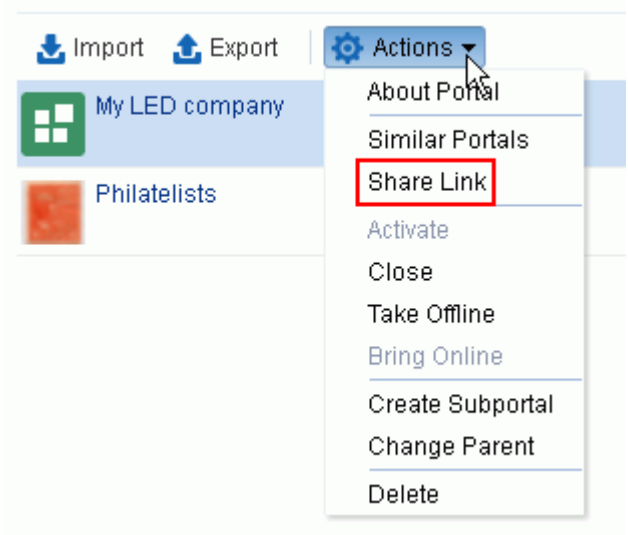
46.8 Sharing the Link to a Portal

If you want to share a portal with others, you can publish a link to the portal that will appear in activity streams of other users. With appropriate permissions, users can directly access a portal by clicking the link that specifies the portal display name.

To publish the direct link to a portal:

1. On the **Portals** page in Portal Builder (see Section 46.2, "Accessing the Portals Page in Portal Builder"), select the required portal by highlighting the row in the table.
2. From the **Actions** menu, select **Share Link** (Figure 46–10).

Figure 46–10 Sharing a Link to a Portal



3. In the Share dialog (Figure 46–11), optionally enter a comment to appear with the link.

Figure 46–11 Share Dialog for a Portal

4. In the **Share with** list, select who you want to share the link with:
   - **Everyone** to share the link with all members of the current portal in their activity streams. This is useful to notify members of updates to the portal.
   - **Portals** to open the Select a Portal dialog, where you can select a portal to share the link in the activity streams of all members of the selected portal. This is useful for sharing information with members of other portals who may be interested in your portal.
5. Click **Publish**.

46.9 Closing Any Portal

By default, a portal is active. You can close a portal that is no longer being actively used. Closing a portal archives its content. When you close a portal, it is removed from everyone's **Portals** menu and displays on the **Portals** page in the Home portal only when a user selects **Closed** from the **Show** list. The content of a closed portal remains accessible and searchable to those who still want to reference it and portal members can continue working in the portal either by displaying closed portals, or by pretty URL (`http://host:port/webcenter/portal/portalName`).

When a portal is closed, any activities performed in the portal are no longer reflected in the Activity Stream in the Home portal. Only the Home page of the closed portal shows activity in the portal.

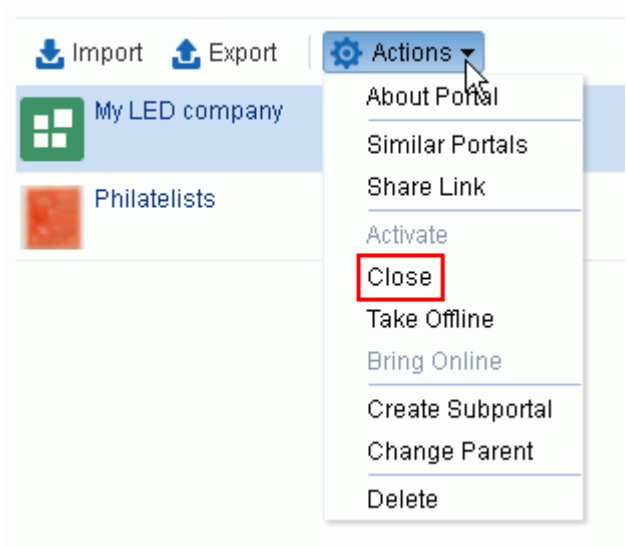
If you want to close down a portal temporarily, take the portal offline instead. See Section 46.11, "Taking Any Portal Offline."

To close a portal:

1. On the **Portals** page in Portal Builder (see Section 46.2, "Accessing the Portals Page in Portal Builder"), select the required portal by highlighting the row in the table.

   Press Ctrl+click to select more than one portal.
2. From the **Actions** menu, select **Close** (Figure 46–12).

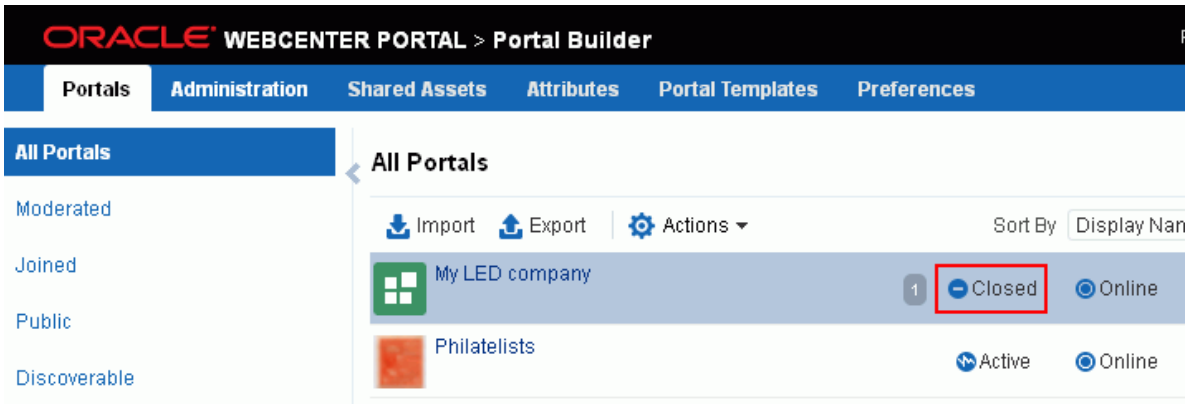
Figure 46–12 Closing a Portal



3. Confirm the action by clicking **OK**.

   Notice that the **Active** status changes to **Closed** (see Figure 46–13).

Figure 46–13 Closed Portal Status



46.10 Reactivating Any Portal

You may close a portal if it is no longer being used (see Section 46.9, "Closing Any Portal"). If you want to reopen a portal, you can reactivate it.

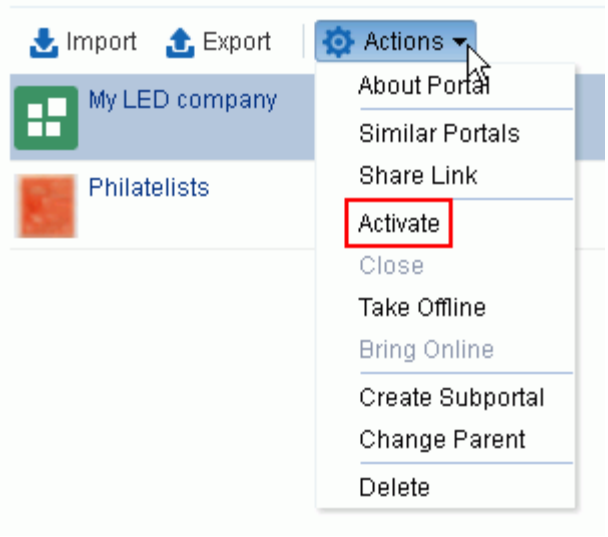
To reactivate a portal:

1. On the **Portals** page in Portal Builder (see Section 46.2, "Accessing the Portals Page in Portal Builder"), select the required portal by highlighting the row in the table.

   Press `Ctrl+click` to select more than one portal.

2. From the **Actions** menu, select **Activate** (Figure 46–14).

Figure 46–14 Activating a Portal



3. Confirm the action by clicking **OK**.

   Notice that the **Closed** status changes to **Active**.

46.11 Taking Any Portal Offline

By default, a portal is online. You can take a portal temporarily offline for maintenance. For example, if you notice inappropriate content, you can take a portal offline to modify its content, then bring it back online. Only the system administrator or portal members with `Manage Configuration` permission can access a portal that is offline, or bring it back online. Other members see the Portal Unavailable page (see Chapter 50, "Customizing System Pages").

To permanently close down a portal that is not being used any more, see Section 46.9, "Closing Any Portal".

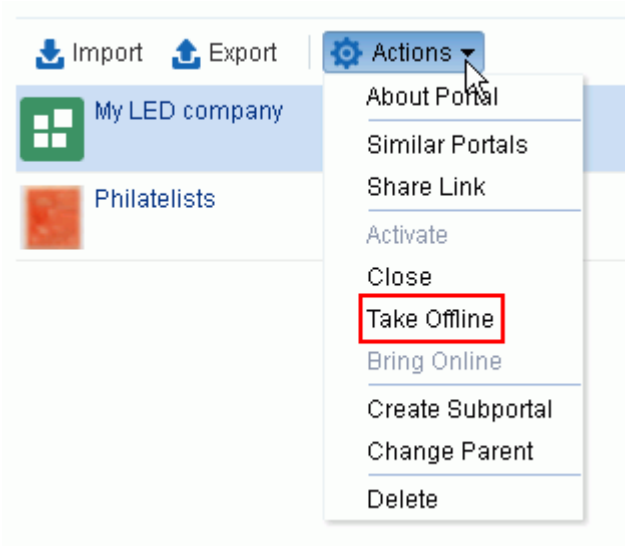
To take a portal offline:

1. On the **Portals** page in Portal Builder (see Section 46.2, "Accessing the Portals Page in Portal Builder"), select the portal you require by highlighting the row in the table.

   Press Ctrl+click to select more than one portal.

2. From the **Actions** menu, select **Take Offline** (Figure 46–15).

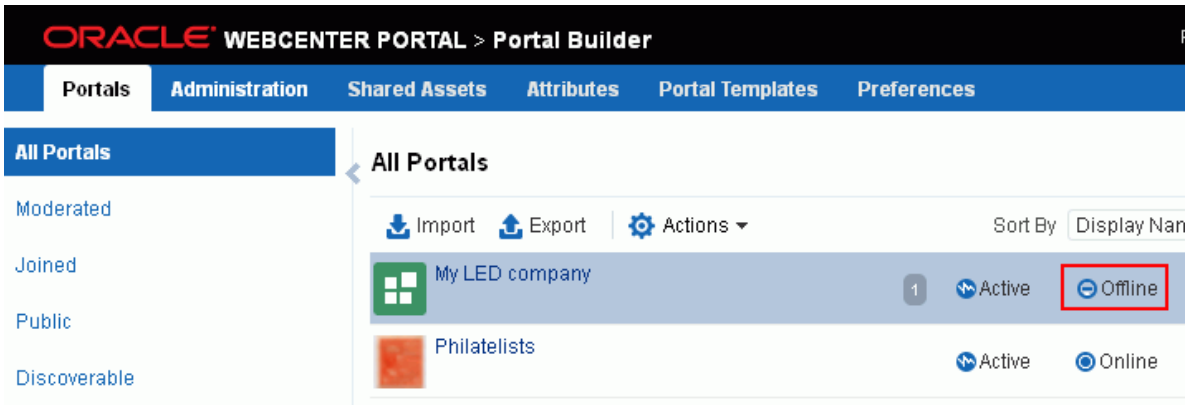
Figure 46–15 Taking a Portal Offline



3. Confirm the action by clicking **OK**.

   Notice that the **Online** status changes to **Offline** (see Figure 46–16).

Figure 46–16 Offline Portal Status



46.12 Bringing Any Portal Back Online

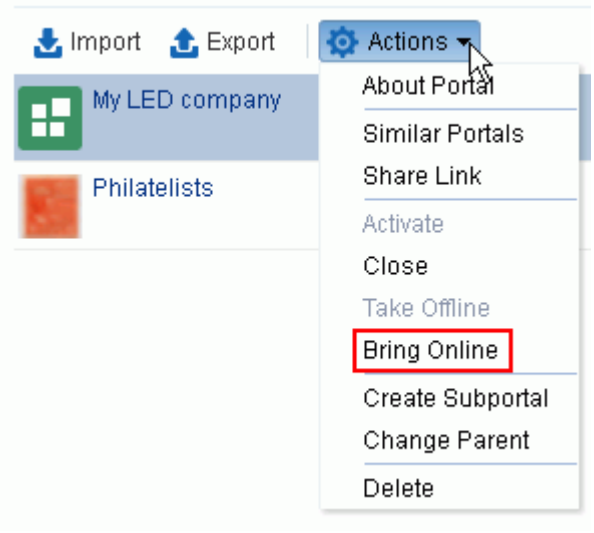
To bring any portal back online:

1. On the **Portals** page in Portal Builder (see Section 46.2, "Accessing the Portals Page in Portal Builder"), select the required portal by highlighting the row in the table.

   Press Ctrl+click to select more than one portal.

2. From the **Actions** menu, select **Bring Online** (Figure 46–17).

Figure 46–17 Bringing a Portal Online



3. Confirm the action by clicking **OK**.

   Notice that the **Offline** status changes back to **Online**.

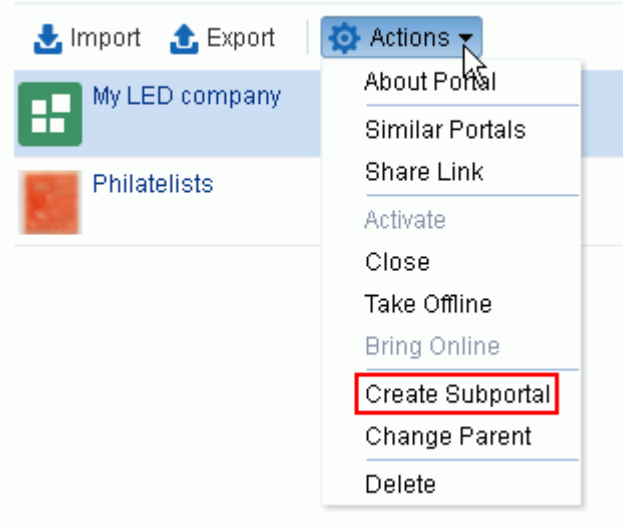
46.13 Creating a Subportal

You can create one or more subportals in any portal. From a parent portal, you can navigate to its subportals (see Section 46.14, "Moving a Portal or Subportal (Changing the Parent)").

To create a subportal:

1. On the **Portals** page in Portal Builder (see Section 46.2, "Accessing the Portals Page in Portal Builder"), click in the row of the portal for which you want to create a subportal.
2. From the **Actions** menu, select **Create Subportal** (Figure 46–18).

Figure 46–18 Creating a Subportal



The **Select a Portal Template** page appears.

For information about creating a subportal, see the "Creating a Subportal" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

46.14 Moving a Portal or Subportal (Changing the Parent)

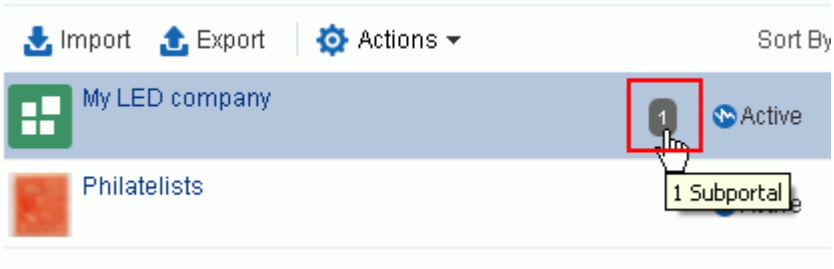
WebCenter Portal system administrators with `Portals-Manage All` permission can move portals to become subportals or change parent for a subportal. All of the metadata (such as pages, navigation, security, and so on) and data is maintained when a subportal is moved.

To move one or more portals subportals from their current parent portal:

1. On the **Portals** page in Portal Builder (see Section 46.2, "Accessing the Portals Page in Portal Builder"), click in the row of the portal or subportal that you want to move.

   To view the subportals for a portal, you can click the **Subportal** icon (Figure 46–19).

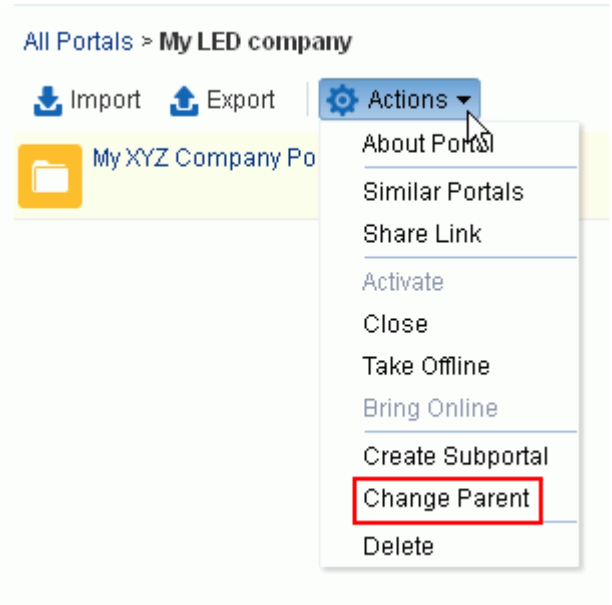
Figure 46–19 Subportal Icon



   Press Ctrl+click to select more than one portal.

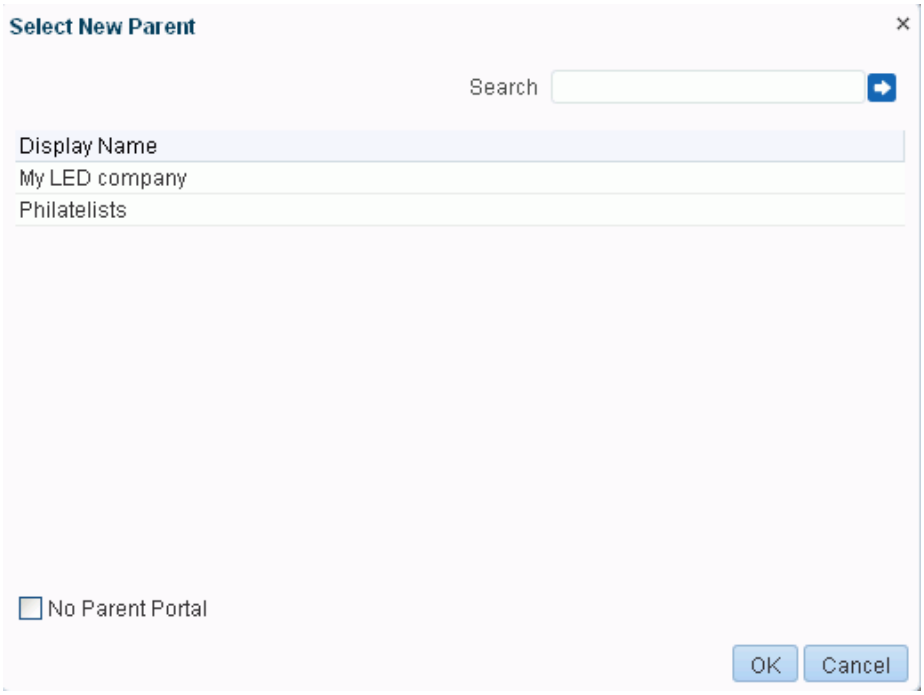
2. From the **Actions** menu, select **Change Parent** (Figure 46–20).

Figure 46–20 Moving a Portal to a Different Parent Portal



3. In the **Select New Parent** dialog (Figure 46–21), select the new parent portal, or select the **No Parent Portal** check box to move the selected portal(s) to the root portal of the portal hierarchy.

Figure 46–21 Select New Parent Dialog



Note: All portals and subportals in WebCenter Portal are listed, which means you can move a selected subportal lower in the portal hierarchy, under another subportal.

4. Click **OK**.

46.15 Deleting a Portal

When a portal has been closed or inactive for some time, you may want to remove it permanently from WebCenter Portal. System administrators with the `Portals-Manage All` permission can delete any portal. Deleting a portal is permanent; it cannot be restored after it is deleted.

When you delete a portal:

- All pages associated with the portal are deleted.
- Links, lists, notes, tags, and events) associated with the portal are deleted.
- Portal roles and membership details are deleted.
- Content managed by discussions and announcements is deleted, when it is stored in the default forum or category created by the portal. Content managed by nondefault forums or categories is not deleted (for details, see the "Modifying Discussion Forum Settings for a Portal" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*).

- The portal mail distribution list that is automatically created by the WebCenter Portal is deleted. However, distribution lists that are customized by the portal moderator are not deleted (for details, see the "Configuring the Mail Distribution List for a Portal" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.
- Content managed by external services, such as content repositories, mail, and so on, is removed.
- If the portal is a parent in a portal hierarchy, child subportals are deleted too.

You cannot delete a portal while the moderator is editing portal settings, but there are no other restrictions.

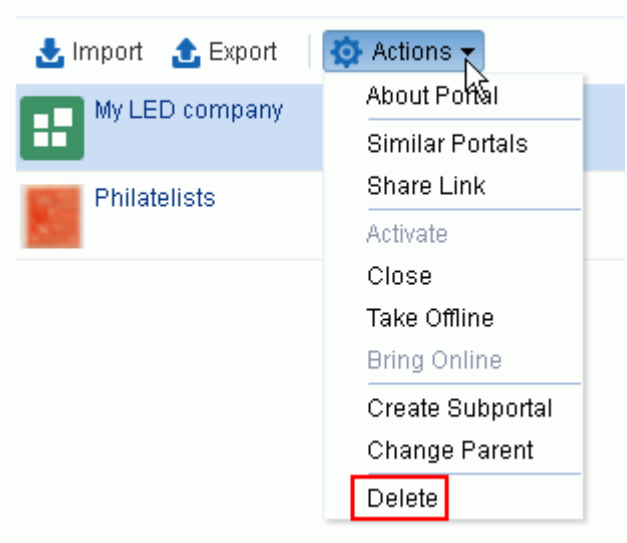
To delete a portal:

1. On the **Portals** page in Portal Builder (see Section 46.2, "Accessing the Portals Page in Portal Builder"), select the portal to delete by highlighting the row in the table.

   Press `Ctrl+click` to select more than one portal.

2. From the **Actions** menu, select **Delete** (Figure 46–22).

Figure 46–22 Deleting a Portal



3. Click **Delete** to confirm that you want to delete the portal(s).

If the delete process fails for any reason, the portal is not removed from your **Portals** tab; this sometimes happens when a back-end server cannot be contacted. If you click **Delete** again, the portal is removed.

47

Exploring the Administration Page in Portal Builder Administration

This chapter describes the Administration page in WebCenter Portal Builder administration and provides an overview of the tasks that are available from the Administration pages.

This chapter includes the following topics:

- Section 47.1, "About Portal Builder Administration"
- Section 47.2, "Accessing the Portal Builder Administration Page"
- Section 47.3, "Performing Portal Builder Administration Tasks"

Permissions: To perform the tasks in this chapter, you must have the WebCenter Portal `Administrator` role or a custom role that grants the following permission:

- `Portal Server-Manage Configuration`

Additionally, you need permissions to create and manage portals (`Portals-Create` and/or `Portals-Manage All`). For more information about permissions, see Section 49.3, "About Application Roles and Permissions."

Notes:

- If you are using Internet Explorer, turn off Compatibility Mode before trying to access WebCenter Portal. In Internet Explorer, from the **Tools** menu, select **Compatibility View Settings**. In the Compatibility View Settings dialog, deselect all the options, and click **Close**.
- WebCenter Portal supports only single browser tab or window viewing. It will not function properly if you try to view WebCenter Portal in multiple browser tabs or windows simultaneously.

47.1 About Portal Builder Administration

From the Portal Builder Administration pages, WebCenter Portal system administrators can administer the entire WebCenter Portal application.

- Users with the `Administrator` role can set application-wide properties for WebCenter Portal, create business role pages, configure defaults for discussion forums, mail, and people connection services, register producers and external applications, as well as perform other administrative duties such as editing the login page and the self-registration page.

  Administrators can also manage users and roles for the WebCenter Portal, delegate or revoke privileges to and from other users, manage portals and portal templates, and also import and export portal information. For more information about the system administrator role, see Section 2.1, "Role of the System Administrator."

- From the WebCenter Portal Administration page, system administrators can specify general settings for WebCenter Portal, such as application name and logo, default skin and page template, resource catalogs, landing pages, and so on; security settings, such as user and group roles and default permissions for each role; manage tools and services; and manage business, system, and personal pages.

  For more information about the administrator tasks that can be accessed from the WebCenter Portal Builder Administration page, see Section 47.3, "Performing Portal Builder Administration Tasks."

47.2 Accessing the Portal Builder Administration Page

Many administrative actions are performed from WebCenter Portal Administration pages. After you log in as the system administrator, you can access this page in the following ways:

- Click the **Administration** link in the menu bar (Figure 47–1).

Figure 47–1 WebCenter Portal Administration Link

- Enter the following URL in your browser to navigate directly to the WebCenter Portal **Administration** page:

  `http://host:port/webcenter/portal/builder/administration`

> **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

47.3 Performing Portal Builder Administration Tasks

In WebCenter Portal, there are seven main Administration pages (Figure 47–2):

- General
- Security
- Tools and Services
- System Pages
- Business Role Pages
- Personal Pages

- Device Settings

Figure 47–2 WebCenter Portal Administration

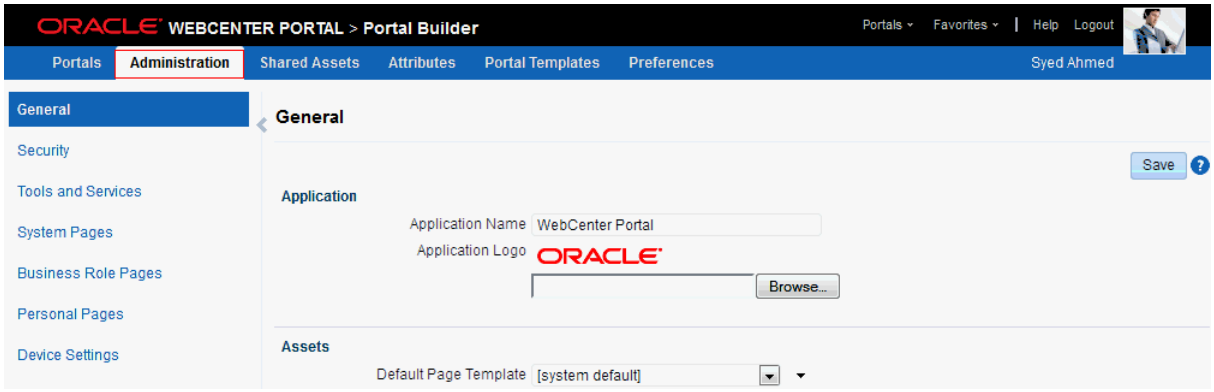


Table 47–1 describes the actions system administrators can perform from the Administration pages and lists the permission required to perform them.

Table 47–1 WebCenter Portal Administration Pages

Page	Description	Required Permission

General	Use this page to set application-level properties for WebCenter Portal, such as:
	▪ application name and logo
	▪ default page template, skin, and navigation
	▪ resource catalogs to use
	▪ page footer options
	▪ default language
	▪ starting page or portal for users and groups
	▪ self-registration options
	For more information, see Chapter 48, "Configuring Global Defaults Across Portals."
		`PortalServer-Manage All`
		or
		`Portal Server-Manage Configuration`

Security	Use this page to view the default security model that enables you to control what users can see and change. You can also add users and groups to WebCenter Portal and assign roles to them. For more information, see Chapter 49, "Managing Security Across Portals."	`Portal Server-Manage All`

Tools and Services	Use this page to manage settings for tools and services in WebCenter Portal.
	For more information, see Chapter 8, "Managing Tools and Services."
		`Portal Server-Manage All`
		or
		`Portal Server-Manage Configuration`

Table 47–1 (Cont.) WebCenter Portal Administration Pages

Page	Description	Required Permission

System Pages	Use this page to customize out-of-the-box preconfigured pages, some of which contain task flows that are available in WebCenter Portal.
	For more information, see Chapter 50, "Customizing System Pages."
		`Portal Server-Manage All`
		or
		`Portal Server-Manage Configuration`
		or
		`Portals-Create, Edit, and Delete Pages`

Business Role Pages	Use this page to work with pages that are targeted to specific users and groups, as well as perform page management tasks for these business role pages.
	For more information, see Chapter 51, "Managing Business Role Pages."
		`Portal Server-Manage All`
		or
		`Portal Server-Manage Configuration`
		or
		`Portals-Create, Edit, and Delete Pages`

Personal Pages	Use this page to manage personal pages that are created by users. Users can create personal pages and set access to these pages. However, as the system administrator, you can edit personal pages created by other users.
	For more information, see Chapter 52, "Managing Personal Pages."
		`Portal Server-Manage All`
		or
		`Portal Server-Manage Configuration`
		or
		`Portals-Create, Edit, and Delete Pages`

Device Settings	Use this page to create and manage device groups and devices for WebCenter Portal.You can create a device group, associate various devices with it, and specify the assets, such as the skin and page template, to be used for the device group.
	For more information, see Chapter 53, "Administering Device Settings."
		`Portal Server-Manage All`
		or
		`Portal Server-Manage Configuration`

48

Configuring Global Defaults Across Portals

This chapter describes the tasks available on the **General** page in WebCenter Portal Builder administration (Figure 48–1). The system administrator can modify the default settings to suit the needs of the organization.

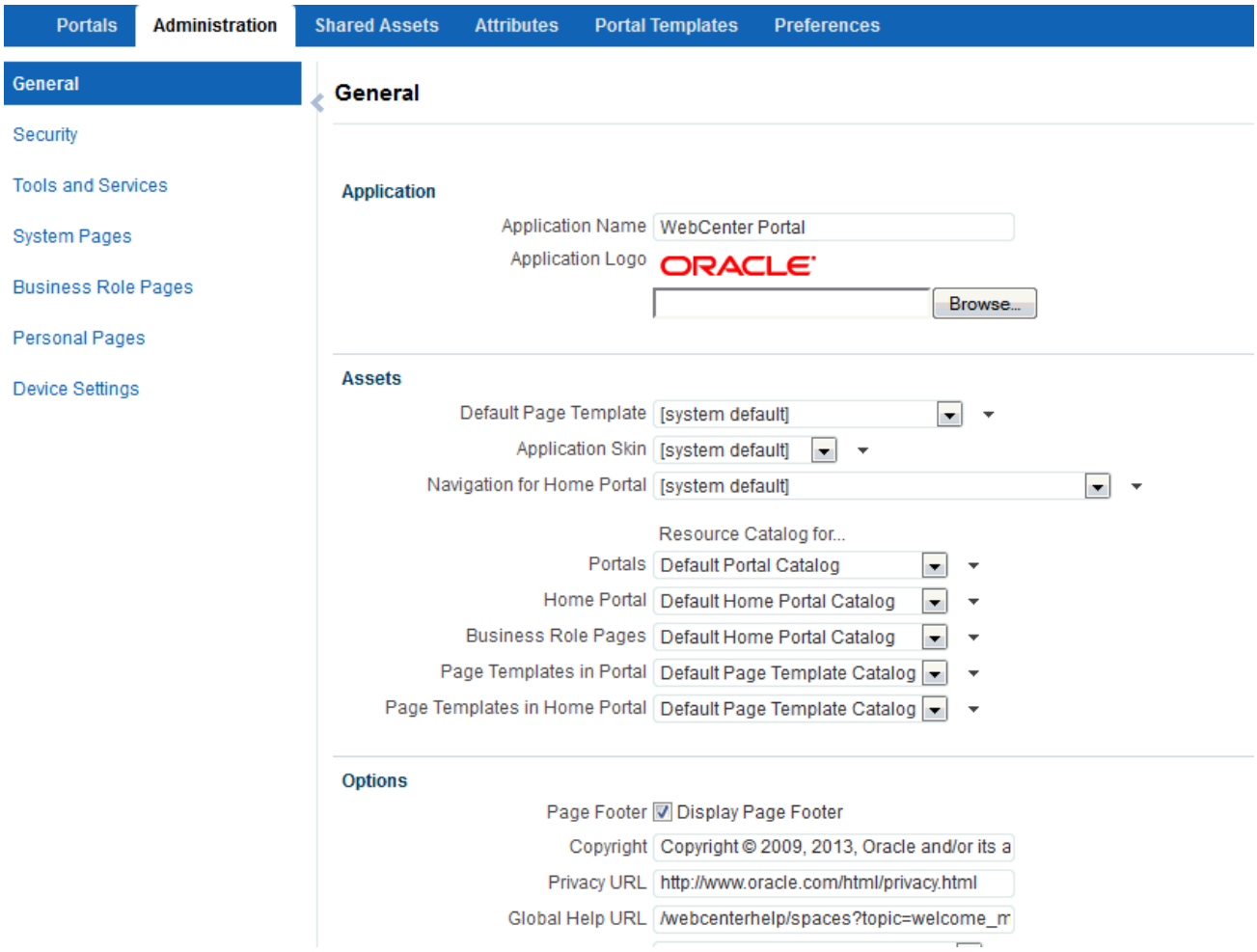
This chapter includes the following topics:

Permissions: To perform the tasks in this chapter, you must have the WebCenter Portal `Administrator` role or a custom role that grants the following permission:

- `Portal Server-Manage Configuration`

Additionally, you need permissions to create and manage portals (`Portals-Create` and/or `Portals-Manage All`). For more information about permissions, see Section 49.3, "About Application Roles and Permissions."

Figure 48–1 Portal Builder Administration: General Page



48.1 Customizing the Name and Logo

Out-of-the-box, the Oracle logo and application name **WebCenter Portal** appear in the banner of all portals (Figure 48–2). You can change both the logo and name to better suit your target audience. For example, you might want to display your company name here or the name of a department within your company.

The logo you specify will resize according to the application's page template. If you want to adjust the logo size, you can modify the page template. See the "Editing a Page Template" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Figure 48–2 WebCenter Portal Name and Logo

To change the name or logo for WebCenter Portal:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **General**.

You can also enter the following URL in your browser to navigate directly to the **General** page:

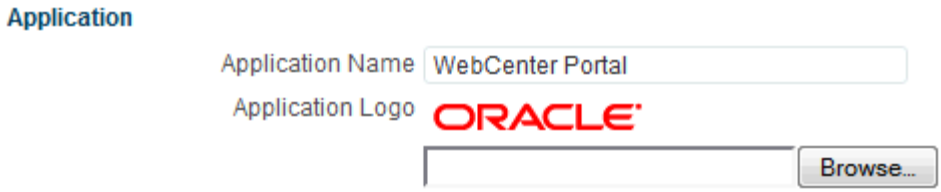
`http://host:port/webcenter/portal/builder/administration/general`

> **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. In the **Application Name** field, enter the new name (Figure 48–3).

   Alphanumeric characters are allowed and also portals, underscores (_) and dashes (-). For example, `Finance Department - My Corporation`.

Figure 48–3 Customizing the Application Name and Logo



3. To change the logo, click **Browse** next to the **Application Logo** field.
4. In the File Upload dialog, navigate to the logo you want to use.

   The logo image file can be up to 150 KB. Supported file formats are `.gif` or `.GIF`, `.png` or `.PNG`, and `.jpg` or `.JPG`. If the file is not uploading, check the size of the file you are trying to upload. The file name must contain alphanumeric characters only.

   The logo is uploaded to WebCenter Portal's image directory (`/webcenter/images`) and the new logo immediately appears in the top left corner of the application banner.
5. Click **Save**.

48.2 Choosing a Default Page Template

In WebCenter Portal, page templates define how individual pages and groups of pages display on a user's screen. Every page displays within a page template. System administrators can define the *default page template* used to display pages in the following places:

- The Home portal
- New portals, when the portal's template does not specify that a particular page template must be used

Portal moderators can override the default selection within their portal, but users cannot override the page template applied to the Home portal.

The Default Page Template for a device group can now also be overridden from **Device Settings** in WebCenter Portal Administration. Select the appropriate Device Group, then select **Edit** from the **Actions** drop-down list. Select the default page template from the **Assets** section for use with devices of the selected group.

Each page template works together with a skin to determine the overall look and feel of the pages in a portal. While the page template controls the location and behavior of components on the page, the skin controls the visual appearance of those components, such as the colors, fonts, and various other aspects.

> **See Also:** For more information about skins, see the "Working with Skins" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Each page template can define a *preferred skin* to identify the skin that works best with that page template. When the page template is selected as the default page template for a portal or as the system default, the default skin automatically updates to the page template's preferred skin.

> **See Also:** For more information, see the "Setting a Page Template's Preferred Skin" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

See also the "Working with Page Templates" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

To select the default page template for WebCenter Portal:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **General**.

   You can also enter the following URL in your browser to navigate directly to the **General** page:

   `http://host:port/webcenter/portal/builder/administration/general`

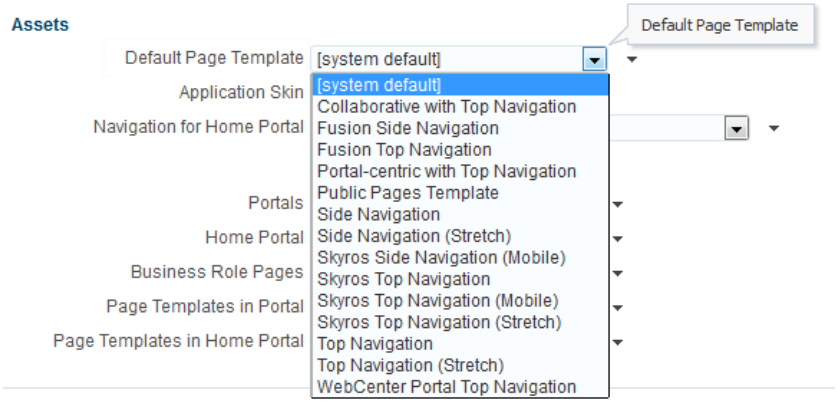
   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Do one of the following:

   - Select a **Default Page Template** from the available list (Figure 48–4).

     To learn how to add page templates to this list, see the "Publishing or Hiding a Page Template" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Figure 48–4 Selecting a Default Page Template



- Click the **Advanced Edit Options** icon, then select **Expression Builder** to enter an EL expression that determines the default page template dynamically based on certain criteria. If you need EL assistance, an application developer can provide an EL expression; see the "Expression Language Expressions" appendix in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

  For example, you may like the default page template to change depending on which department or organization the logged in user belongs to.

3. Click **Save**.

48.3 Choosing a Default Skin

As a system administrator, you may customize the default appearance of WebCenter Portal for all users by changing the default skin. A skin changes the way the user interface appears, but does not change the application's behavior. See Section 48.3.1, "Applying a Skin for WebCenter Portal."

Users can override the default skin selection through user preferences. However, skins are often created for use with a specific page template. The choice of skin must therefore be compatible with the selected page template. For more information, see the "Changing the Look and Feel of Your View" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

If none of the built-in skins suit your requirements or you want to apply a look and feel that reflects your corporate brand, you can create and apply your own ADF skins. For your own page templates, you can note the *preferred skin* by setting (select **Shared Assets**, then copy a Page Template and select **Edit Properties** from the **Actions** drop-down list) the custom attribute `preferredSkin` to the skin family ID value of the skin that is preferred for use with a given page template. Doing this will allow the skin to switch to the preferred skin when your page template is chosen. If the page template is changed, then the skin will be updated (if it is not set to an expression) to match the page template. For more information, see the "Creating a Skin" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

The Default Skin for a device group can now also be overridden from **Device Settings** in WebCenter Portal Administration. Select the appropriate Device Group, then select **Edit** from the **Actions** drop-down list. Select the default skin from the **Assets** section for use with devices of the selected group.

If you want, you can reference the default skin in EL expressions. If you need EL assistance, an application developer can provide an EL expression; see the "Expression Language Expressions" appendix in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

Individual users can change the skin applied to their Home portal view through user preferences if they do not like the default skin that you specify. See the "Setting the Default Skin for a Portal" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

48.3.1 Applying a Skin for WebCenter Portal

When you set a skin for WebCenter Portal, the skin is applied to the Home portal and all portals that use the application-level skin setting. The skin is also applied to any new portals that are created.

To apply a skin to WebCenter Portal:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **General**.

   You can also enter the following URL in your browser to navigate directly to the **General** page:

   `http://host:port/webcenter/portal/builder/administration/general`

   See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Do one of the following:

   - Select an **application skin** from the available list (Figure 48–5).

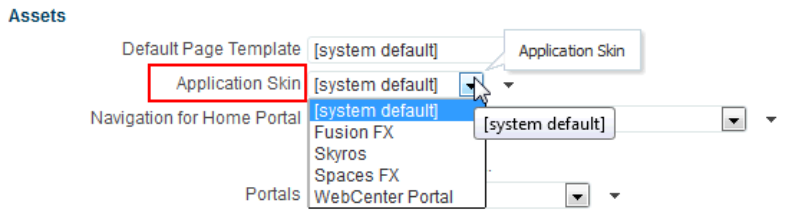
     Note: If the desired skin does not appear in the **Application Skin** list, it may be because its Available option has been deselected. For information, see the "Managing a Skin" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

     Each page template can define a *preferred skin* to identify the skin that works best with that page template. When a page template is selected as the new default page template for a portal or as the system default, the default skin automatically updates to the page template's preferred skin.

     See Also: For more information, see the "Setting a Page Template's Preferred Skin" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

     WARNING: changing the default skin to something other than the preferred skin for the selected default page template may produce unexpected results.

Figure 48–5 Applying a Skin to WebCenter Portal



   - Click the **Advanced Edit Options** icon, then select **Expression Builder** to enter an EL expression that determines the default application skin dynamically based on certain criteria.

     For example, you may like the default skin to change depending on which department or organization the logged in user belongs to.

3. Click **Save**.

   The skin you select is applied to WebCenter Portal, any new portals that are created, and all portals that use the application-level skin setting. The skin is not applied to the portals that override the application-level skin setting.

48.4 Choosing a Default Navigation

In WebCenter Portal, navigation models allow users to see and navigate to information quickly and easily. System administrators can specify which navigation to use for presenting Home portal information.

Alternatively, you can enter an EL expression that determines the navigation dynamically based on certain criteria. If you need EL assistance, an application developer can provide an EL expression; see the "Expression Language Expressions" appendix in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*. For example, you may like to display different navigations depending on which department or organization the logged in user belongs to.

See also the "Working with Navigations" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

To select a navigation model for the Home portal:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **General**.

   You can also enter the following URL in your browser to navigate directly to the **General** page:

   `http://host:port/webcenter/portal/builder/administration/general`

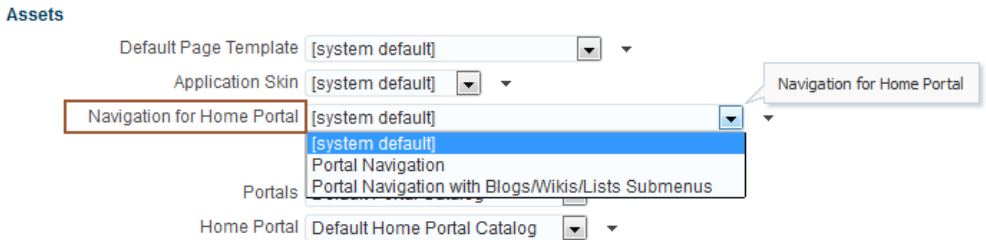
   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Do one of the following:

   - Select a **Navigation for Home Portal** from the available list (Figure 48–6).

     To learn how to add navigations to this list, see the "Showing and Hiding Portal Assets" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Figure 48–6 Selecting a Navigation for Home Portal



   - Click the **Advanced Edit Options** icon, then select **Expression Builder** to enter an EL expression that determines the navigation dynamically based on certain criteria. If you need EL assistance, an application developer can provide an EL expression; see the "Expression Language Expressions" appendix in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

3. Click **Save**.

48.5 Choosing Default Resource Catalogs

In WebCenter Portal, a series of Resource Catalogs display when you edit a page or a page template in Composer and click **Add Content**. Each catalog presents available resources in a series of folders and subfolders and the content changes dynamically depending on which services are currently available and the permissions of the person who is editing the page or page template. Available resources include task flows, portlets, and page components, such as images, content boxes, hyperlinks, and the like.

WebCenter Portal provides several default resource catalogs out-of-the-box, but you can add new task flows, remove task flows, or reorganize the folder hierarchy to better suit your audience or create brand new ones of your own. For details, see the "Working with Resource Catalogs" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

System administrators can specify *default page catalogs* for:

- New portals
- Home portal
- Business role pages

As well as, *default page template catalogs* for:

- New portals
- Home portal

It is entirely up to you whether you specify a different catalog for each scenario or use the same catalog throughout.

To select default resource catalogs for WebCenter Portal:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **General**.

   You can also enter the following URL in your browser to navigate directly to the **General** page:

   `http://host:port/webcenter/portal/builder/administration/general`

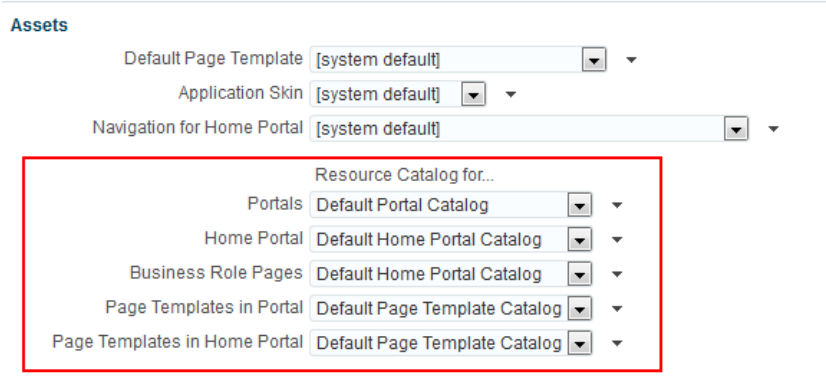
   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. In the Assets section, select default resource catalogs for the following (Figure 48–7):

   - Portals
   - Home portal
   - Business Role Pages
   - Page Templates in Portal
   - Page Templates in Home portal

   To learn how to add resource catalogs to this list, see the "Showing and Hiding Portal Assets" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Figure 48–7 Selecting Resource Catalogs for Assets



3. Optionally, click the **Advanced Edit Options** icon, then select **Expression Builder** to enter an EL expression that determines the default resource catalog dynamically based on certain criteria. For example, you may like the default resource catalog to change depending on the which role the logged in user belongs to. If you need EL assistance, an application developer can provide an EL expression; see the "Expression Language Expressions" appendix in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.
4. Click **Save**.

48.6 Customizing Copyright and Privacy Statements

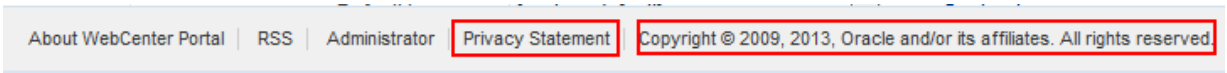
System administrators can customize or hide copyright and privacy statements in WebCenter Portal:

- Copyright - Displays a copyright statement for the entire application.
- Privacy URL - Links to a document that contains a privacy policy for the entire application.

In the default page template, the copyright and privacy URL appear in the WebCenter Portal's page footer (Figure 48–8).

Optionally, you can reference your copyright message and privacy document in EL expressions. If you need EL assistance, an application developer can provide an EL expression; see the "Expression Language Expressions" appendix in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

Figure 48–8 Copyright and Privacy Statements in Page Footer



Individual portals may provide their own copyright and privacy statements. For details, see the "Customizing the Copyright Statement and Privacy URL" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

To customize or hide copyright and privacy statements:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **General**.

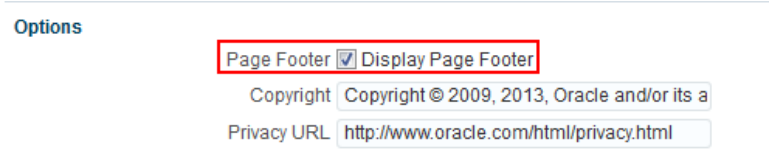
   You can also enter the following URL in your browser to navigate directly to the **General** page:

```
http://host:port/webcenter/portal/builder/administration/general
```

See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. In the Options section, select or deselect **Display Page Footer** to, respectively, display or hide copyright and privacy information in the page footer (Figure 48–9).

Figure 48–9 Customizing the Copyright and Privacy URL



Modify the legal notice and privacy URL as appropriate:

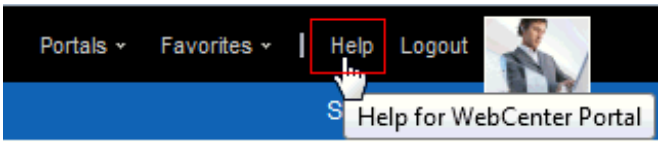
- **Copyright** - Enter a suitable copyright statement for WebCenter Portal. If no copyright information is required, leave this field blank.
- **Privacy URL** - Specify the location of the application's privacy policy. Enter a fully qualified URL. If no privacy information is required, leave this field blank.

3. Click **Save**.

48.7 Customizing the Online Help Link

Online help for WebCenter Portal displays when you click the **Help** link (Figure 48–10).

Figure 48–10 Help Link for WebCenter Portal



Out-of-the-box, this **Help** link opens Oracle's built-in help. See the "Global Help" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*. You can also write online help specifically aimed at your users and redirect the Help link to a different help location.

Optionally, you can reference the Help location in EL expressions. If you need EL assistance, an application developer can provide an EL expression; see the "Expression Language Expressions" appendix in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

When you customize the Help link, built-in help for WebCenter Portal is still available through help buttons, help icons, and so on.

To customize the main Help link:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **General**.

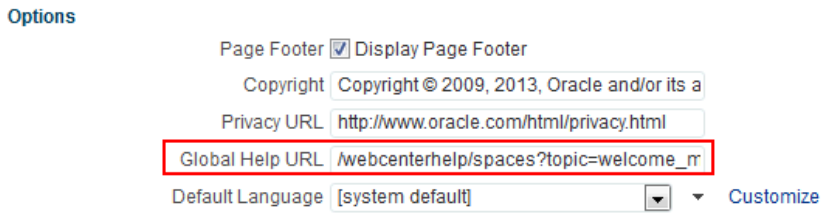
   You can also enter the following URL in your browser to navigate directly to the **General** page:

`http://host:port/webcenter/portal/builder/administration/general`

> **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. In the Options section, enter the location of your help in the **Global Help URL** field (Figure 48–11):

Figure 48–11 Global Help URL for WebCenter Portal Online Help



Ensure that you enter a fully qualified URL in the format:

`http://host:port/helplocation`

For example:

`http://myhost:8888/myhelp`

The default Global Help URL is `/webcenterhelp/spaces?topic=welcome_main`. This URL opens Oracle Help for the Web (OHW) and displays Oracle's built-in help for WebCenter Portal. Enter this URL if you want to return to the default setting.

> **Note:** If you leave the **Global Help URL** field blank, the **Help** link is not displayed.

3. Click **Save**.

Click **Help** in WebCenter Portal to check whether your custom help opens correctly.

48.8 Choosing a Default Display Language

Out-of-the-box, WebCenter Portal supports 27 languages and 100 different locales. It is the system administrator's job to choose a default display language for WebCenter Portal. When selecting the default language, consider which language suits the majority of people using the application. Alternatively, enter an EL expression that determines the default language dynamically based on certain criteria. For example, you may prefer the default display language to change according to the location or organization of the user that is logged in. If you need EL assistance, an application developer can provide an EL expression; see the "Expression Language Expressions" appendix in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

The first time a user logs in to WebCenter Portal the default language displays, but individuals can personalize their display language through user preferences. See the "Setting a Portal Display Language" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

The default display language only applies when users log in to WebCenter Portal. Public pages, such as the welcome page and the login page, display in the browser language. If no default language is provided, the browser language is used. See also the "Display Language Precedence" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Note: Moderators can nominate a display language for a particular portal. When defined, the portal language overrides both the default language and any user language preference. See also the "Setting a Portal Display Language" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

To select the default display language for WebCenter Portal:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **General**.

   You can also enter the following URL in your browser to navigate directly to the **General** page:

   `http://host:port/webcenter/portal/builder/administration/general`

   See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Do one of the following:

   - Select a **Default Language** from the list available (Figure 48–12).

     If the language you want is not available in the drop-down list, click **Customize**, select the check box for the language you require, and click **Save**. See Section 48.8.1, "Customizing the Language List."

Figure 48–12 Selecting a Default Language

     To add a completely new language, your localization team must translate WebCenter Portal resource bundles into the new language, and then these translations must be deployed to the managed server on which WebCenter Portal is deployed. For details, see Section 45.5, "Adding Support for a New Language to WebCenter Portal."

   - Click the **Advanced Edit Options** icon, then select **Expression Builder** to enter an EL expression that determines the default language dynamically based on certain criteria. If you need EL assistance, an application developer can provide an EL expression; see the "Expression Language Expressions" appendix in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

3. Click **Save**.

The display languages that are available for selection here are also offered to users and portal moderators through user preferences. As the system administrator, you can reduce the range of languages available to users. For details, see Section 48.8.1, "Customizing the Language List."

48.8.1 Customizing the Language List

Out-of-the-box, WebCenter Portal offers 27 languages and 100 different locales and all these languages are available to users by default. As the system administrator, you can tailor the languages that are offered to suit your audience. For example, you may prefer to remove all the territory language variants in favor of a more simplified language list or only offer European languages if your portal is specifically aimed at a European audience.

To customize the languages available to users:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **General**.

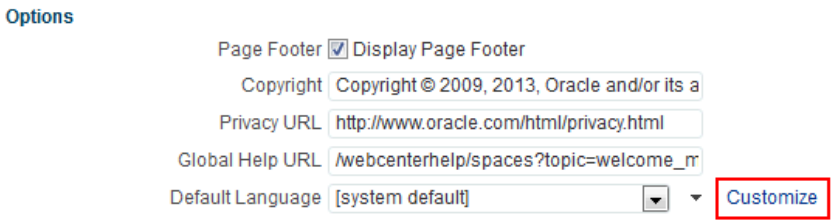
   You can also enter the following URL in your browser to navigate directly to the **General** page:

   `http://host:port/webcenter/portal/builder/administration/general`

   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

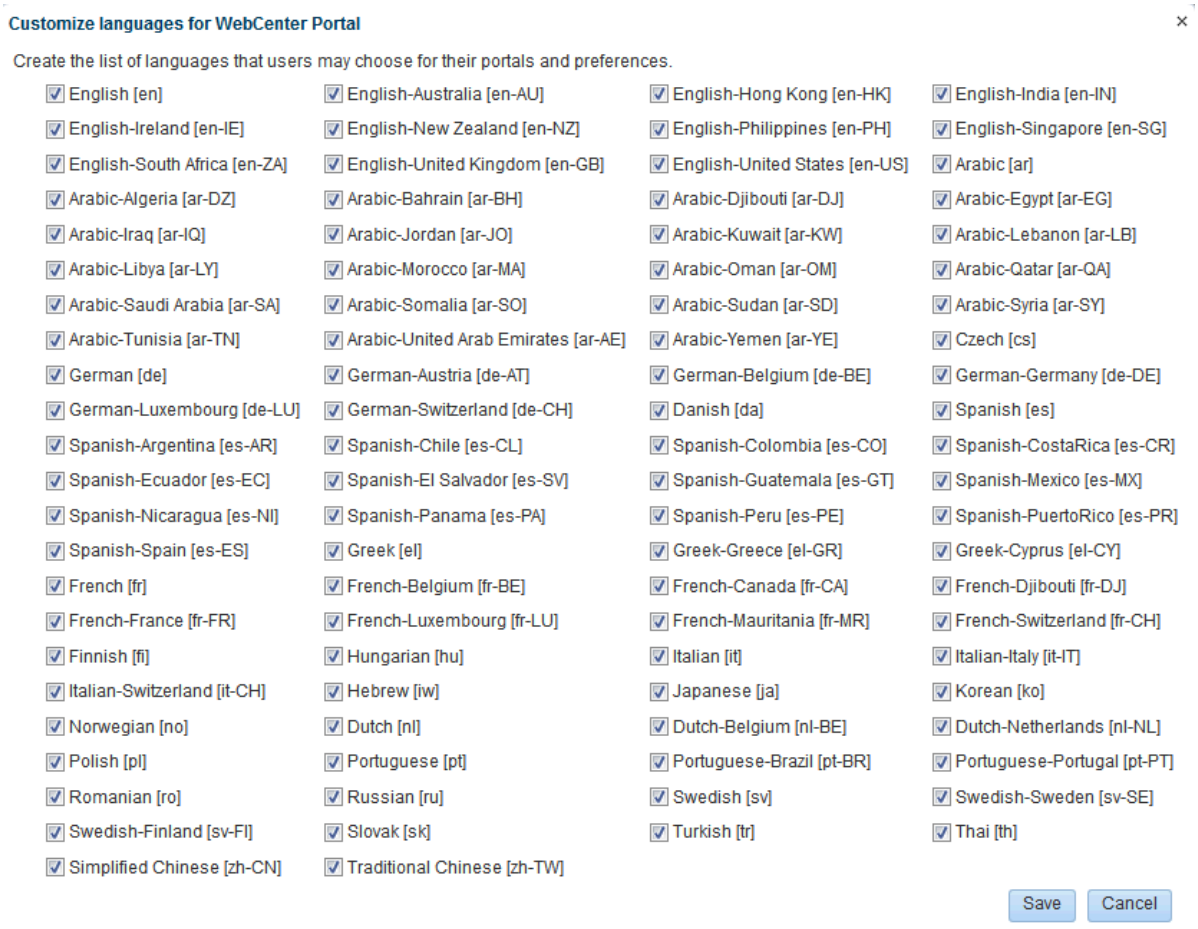
2. Click **Customize** next to **Default Language** (Figure 48–13).

Figure 48–13 Customize Option for Default Language



3. Select which languages to offer by selecting (or deselecting) each language check box (Figure 48–14).

Figure 48–14 Selecting Which Languages Are Available



4. Click **Save**.

48.9 Choosing a Default Start (or Landing) Page

By default, users see the Home portal when they log in, but you can configure the initial landing page to be a specific portal or page. You can specify a start page for a specific group, for authenticated users, and for public users. For more information, see the "Managing Roles and Permissions for a Portal" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

To select the landing page for WebCenter Portal:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **General**.

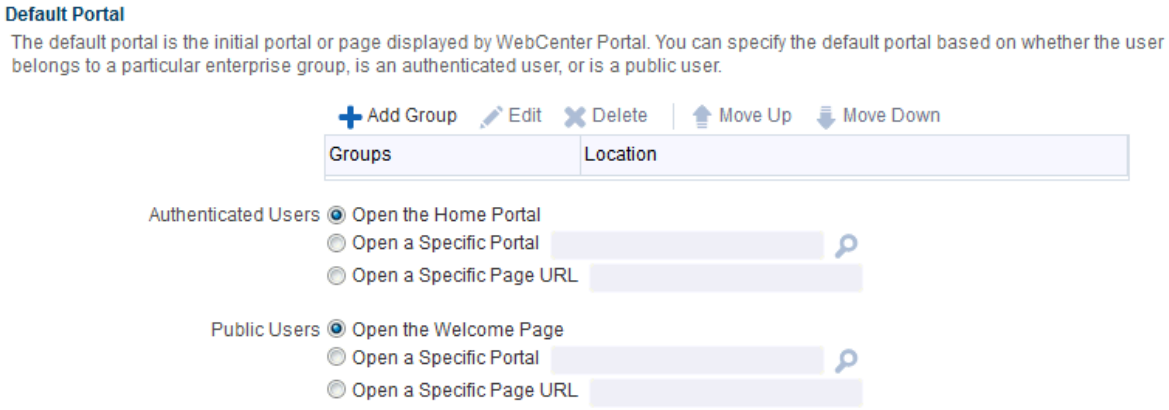
   You can also enter the following URL in your browser to navigate directly to the **General** page:

   `http://host:port/webcenter/portal/builder/administration/general`

   See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. In the **Default Portal** section, select what users see first when they log in: the Home portal, or a specific portal or page (Figure 48–15).

Figure 48–15 Choosing a Default Start Page



- To specify a default landing page for selected groups, see Section 48.9.1, "Choosing a Default Start Page for Groups."
- To specify a default landing page for all other authenticated users who do not belong to any of the specified groups, see Section 48.9.2, "Choosing a Default Start Page for Authenticated Users."
- To specify a default landing page for all public users, see Section 48.9.3, "Choosing a Default Start Page for Public Users."

3. Click **Save**.

48.9.1 Choosing a Default Start Page for Groups

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **General**.

   You can also enter the following URL in your browser to navigate directly to the **General** page:

   `http://host:port/webcenter/portal/builder/administration/general`

   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

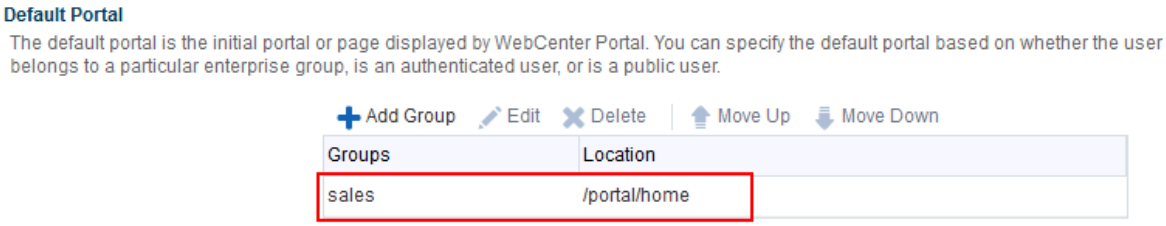
2. In the **Default Portal** section, click **Add Group** if you want selected enterprise groups to see a specific start page (Figure 48–15).

   > **Note:** For the Default portal to be visible to a group member, the group itself should be a member of the portal, if the portal is hidden or private.

3. From the Add Group dialog, search for a group or select a group from the list, then click **OK**.

   The selected group appears (Figure 48–16).

Figure 48–16 Specifying a Landing Page for a Group

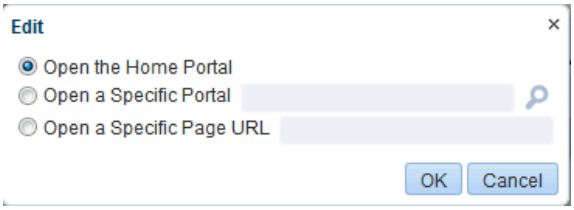


Notice that the group will go to the Home portal upon logging in.

4. To change the **Location** of the landing page, select the group name and click **Edit**.

   The Edit dialog appears (Figure 48–17)

Figure 48–17 Landing Page Options



5. Select whether the group will first see the Home portal, or a specific portal or page:

   - **Open the Home Portal**. Selected by default. Users see the Home portal when they first log in.
   - **Open a Specific Portal**. Select to specify that a particular portal displays, and enter the portal name or click **Browse** to select from a list of portals. Select an option from the **Show Portals** list, and click **OK**. For example,

     `http://host:port/webcenter/portal/portalName`

   - **Open a Specific Page URL**. Select to specify that a particular page displays, and enter the page location.

     Typically this is an internal page. You can enter a full or relative page URL as shown in these examples:

     `http://mywebcenter.com:8888/webcenter/portal/page/landingpage`

     `http://mywebcenter.com:8888/webcenter/portal/portalname/page/landingpage`

     `/portals/portalname/page/landingpage`

     If you specify an external page, make sure that you specify the full URL.

6. Click **Save**.

48.9.2 Choosing a Default Start Page for Authenticated Users

WebCenter Portal users have the default `Authenticated-User` role. A portal is not fully accessible until you give the `Authenticated-User` role permissions to access the pages of the portal. For more information, see the "Viewing and Editing

Permissions for a Portal Role" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **General**.

   You can also enter the following URL in your browser to navigate directly to the **General** page:

   `http://host:port/webcenter/portal/builder/administration/general`

   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. In the **Authenticated Users** section, specify what authenticated users, who are not in any of the groups, specified in Section 48.9.1, "Choosing a Default Start Page for Groups," see when they first log in (Figure 48–18).

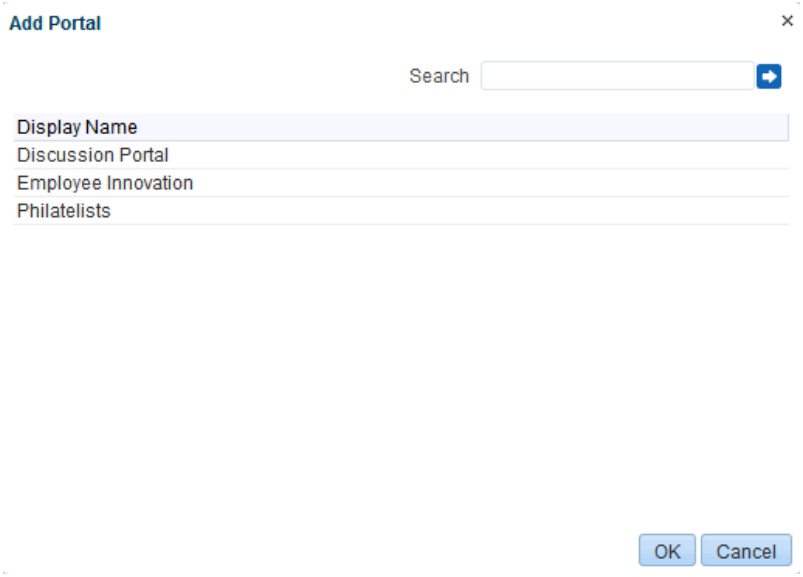
Figure 48–18 Selecting a Landing Page for Authenticated Users

- **Open the Home Portal**. Selected by default. Users see the Home portal when they first log in.
- **Open a Specific Portal**. Select to specify that a particular portal displays, and enter the portal name For example:

  `http://host:port/webcenter/portal/portalName`

  Or click **Browse** to select from a list of portals (Figure 48–19). Select an option from the **Add Portal** list, and click **OK**, or enter the portal name in the **Search** field and click **Search**.

Figure 48–19 Add Portal Dialog



- **Open a Specific Page URL**. Select to specify that a particular page displays, and enter the page location.

  Typically this is an internal page. You can enter a full or relative page URL as shown in these examples:

  ```
  http://mywebcenter.com:8888/webcenter/portal/page/landingpage
  ```

  ```
  http://mywebcenter.com:8888/webcenter/portal/portalname/page/landingpage
  ```

  ```
  /portals/portalname/page/landingpage
  ```

  If you specify an external page, make sure that you specify the full URL.

3. Click **Save**.

48.9.3 Choosing a Default Start Page for Public Users

Any user with access to WebCenter Portal who is not logged in assumes the `Public-User` role. For more information, see the "Managing Roles and Permissions for a Portal" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

You can make a portal available to anyone with access to the WebCenter Portal instance that contains the portal. Registering for a WebCenter Portal account is not required. The public information provided allows the portal to be shared with non-members and people outside of the WebCenter Portal community. For more information, see the "Granting Public Access to a Portal" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **General**.

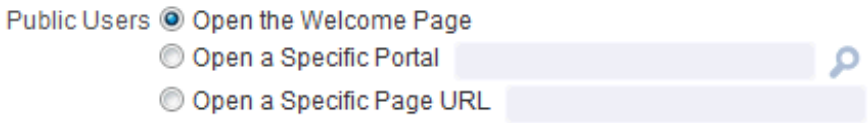
   You can also enter the following URL in your browser to navigate directly to the **General** page:

   ```
   http://host:port/webcenter/portal/builder/administration/general
   ```

   See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. In the **Default Login Settings** section, specify what public users see when they first log in (Figure 48–20).

Figure 48–20 Selecting a Landing Page for Public Users



- **Open the Home Portal**. Selected by default. Users see the Home portal when they first log in.
- **Open a Specific Portal**. Select to specify that a particular portal displays, and enter the portal name. For example,

  ```
  http://host:port/webcenter/portal/portalName
  ```

Or click **Browse** to select from a list of portals (Figure 48–19). Select an option from the **Add Portal** list, and click **OK**, or enter the portal name in the **Search** field and click **Search**.

- **Open a Specific Page URL**. Select to specify that a particular page displays, and enter the page location.

  Typically this is an internal page. You can enter a full or relative page URL as shown in these examples:

  ```
  http://mywebcenter.com:8888/webcenter/portal/page/landingpage
  ```

  ```
  http://mywebcenter.com:8888/webcenter/portal/portalname/page/landingpage
  ```

  ```
  /portals/portalname/page/landingpage
  ```

  If you specify an external page, make sure that you specify the full URL.

3. Click **Save**.

48.10 Specifying Session Timeout Settings

When there is no activity for an extended period of time in a WebCenter Portal session, it times out. Out-of-the box, the WebCenter Portal session timeout is set to 45 minutes. You can modify the default number of minutes that can elapse before a session times out, and select whether you want to display a popup or a window when the session times out.

Note: The value for session timeout can also be seen on the **Attributes** page with the attribute name `wcSessionTimeoutPeriod`.

To modify the session timeout settings for WebCenter Portal:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **General**.

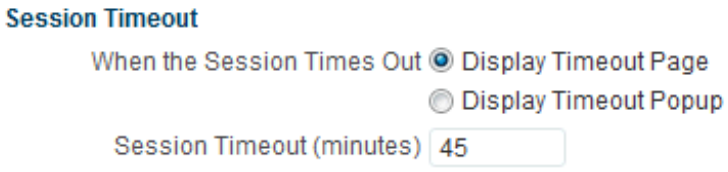
   You can also enter the following URL in your browser to navigate directly to the **General** page:

   `http://host:port/webcenter/portal/builder/administration/general`

   See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

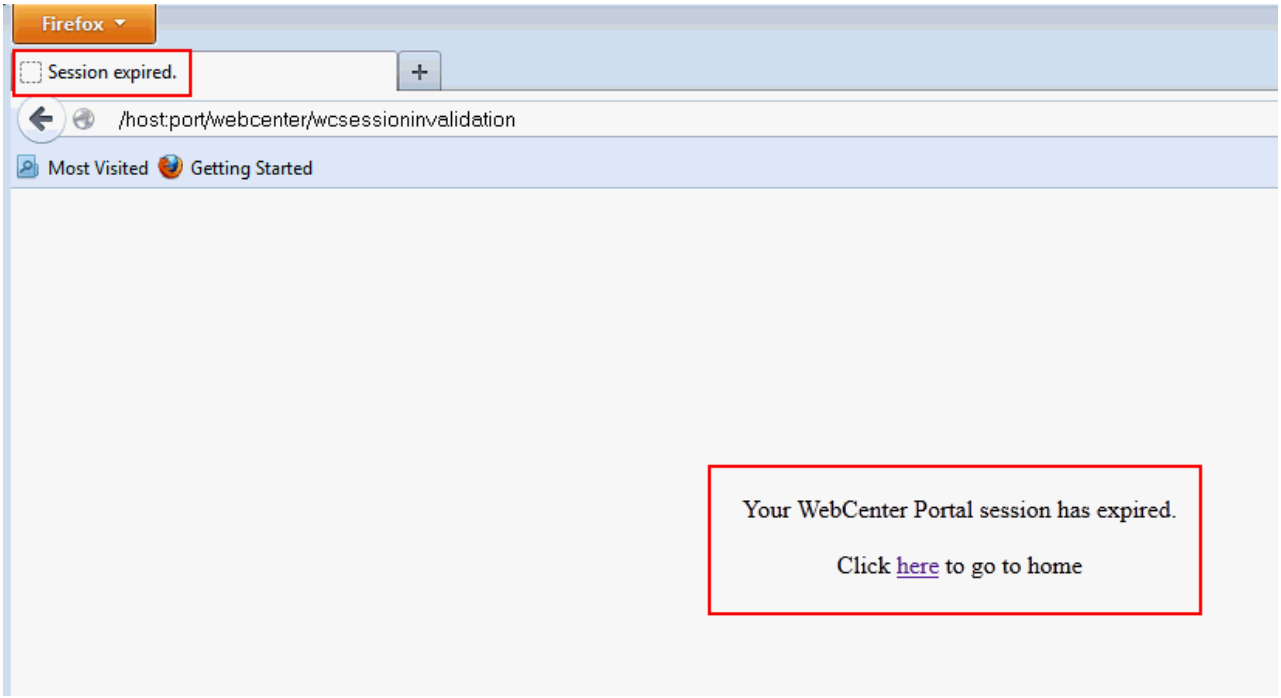
2. In the **Session Timeout** section, select the desired result when WebCenter Portal times out (Figure 48–21).

Figure 48–21 Session Timeout Options



- **Display Timeout Page**. Select to display the WebCenter Portal timeout page in the browser (Figure 48–22), where the user can click the provided link to log in again and restart at the default start page (see also Section 48.9, "Choosing a Default Start (or Landing) Page").

Figure 48–22 Timeout Page



- **Display Timeout Popup**. Select to display an Expiration Warning notification popup (Figure 48–23) when the Session Timeout value is reached. The user can click **OK** within 2 minutes to prevent the timeout. If the user does not respond to the Expiration Warning within 2 minutes, then the session times out. In the Timeout notification popup (Figure 48–24), the user can click **OK** to log in again and restart at the page that was active when the session expired.

  The Display Timeout Popup option works if your browser is set to display popups. If your browser is set to block popups, then you see the timeout page.

Figure 48–23 Expiration Warning Notification (displays at Session Timeout)

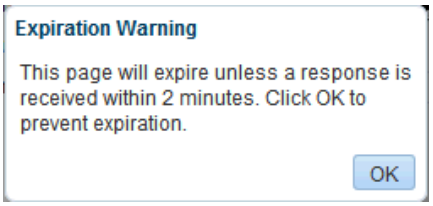
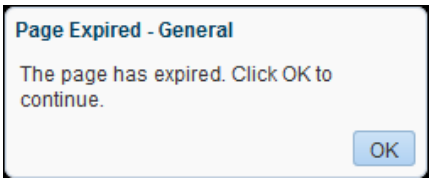


Figure 48–24 Timeout Notification (displays 2 minutes after Session Timeout)



3. (Optional) In the **Session Timeout (minutes)** field, enter a new value.

   The default value is 45 minutes, the minimum value is 5, and the maximum value is 1440 (24 hours). If this field is left blank, the default value (45) applies.

   Note: If the WebCenter Portal is configured for single sign-on (SSO), Oracle recommends that the Session Timeout value set here is no higher than the SSO timeout value. The session timeout is a factor of the physical memory available and the number of concurrent users that have to be supported. If the Session Timeout value is less than the SSO session timeout, then the WebCenter HTTP session times out after the duration specified here, but a new WebCenter session will be automatically created as long as the SSO timeout is not reached.

4. Click **Save**.

48.11 Enabling Self-Registration

WebCenter Portal system administrators can enable self-registration for the application. Through self-registration, invited and uninvited users can create their own login and password. A user who self-registers is immediately and automatically granted access to WebCenter Portal and a new user account is created in the identity store.

This section includes the following information:

- Section 48.11.1, "About Self-Registration"
- Section 48.11.2, "Enabling Anyone to Self-Register"
- Section 48.11.3, "Enabling Self-Registration By Invitation-Only"

48.11.1 About Self-Registration

Self-registration allows users to create their own login and password for WebCenter Portal. A user who self-registers is immediately and automatically granted access to WebCenter Portal and a new user account is created in WebCenter Portal's identity store.

If *anyone* is allowed to self-register, that is any public user, a **Register** link or button displays on the WebCenter Portal login page. To enable this feature, see Section 48.11.2, "Enabling Anyone to Self-Register."

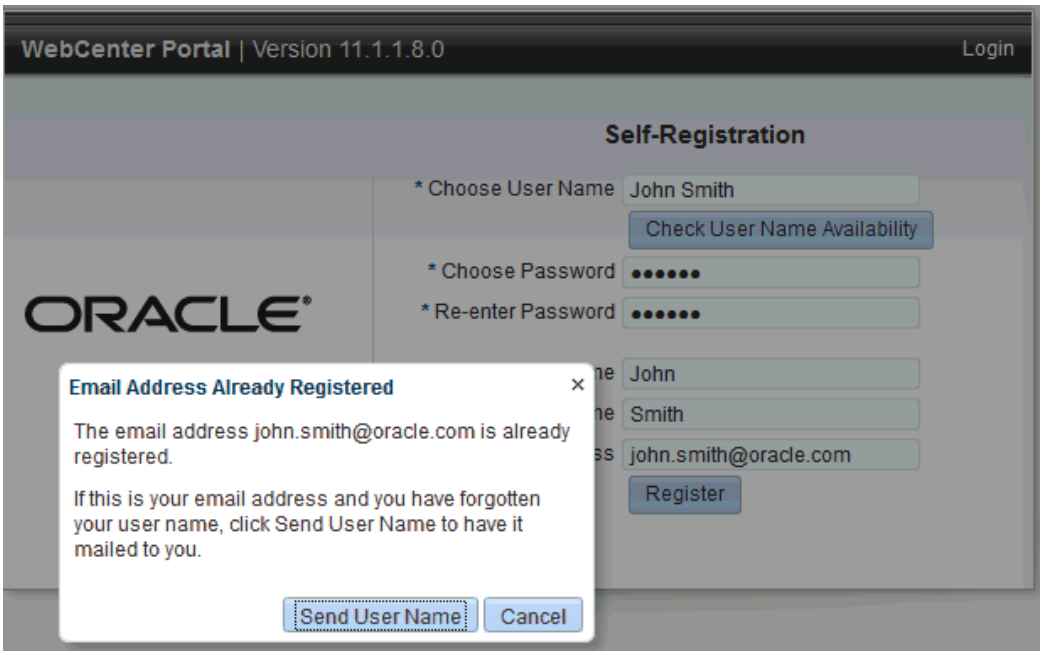
Self-registration by invitation is available too. This feature allows portal moderators to send out membership invitations to people who are not currently registered with WebCenter Portal but might be interested in their portal. Before accessing the portal, invitees must create an account with WebCenter Portal and their account details are added to WebCenter Portal's identity store. If approval is required, the moderator must approve their subscription request before gaining access to the portal. If the portal is public or further approval is not required, then new user gains access to the portal immediately. See also Section 48.11.3, "Enabling Self-Registration By Invitation-Only."

Note: If self-registration is not enabled for WebCenter Portal, identity store management takes place through the WLS Administration Console (or directly into embedded LDAP identity stores using LDAP commands) and is the responsibility of your systems administrator. See also "Section 31.3, "Adding Users to the Embedded LDAP Identity Store."

A self-registration page is supplied out-of-the-box in the System Pages page. Users with the `Administrator` role can add new components to the page and change the page layout if required. See Chapter 50, "Customizing System Pages."

The self-registration page provided with WebCenter Portal offers to send a "user name reminder email" to anyone who tries to register with an email address that has already been used (Figure 48–25).

Figure 48–25 Email Address Already Registered



This feature only works if *public credentials* are defined for the external application that is providing authentication for the mail service in WebCenter Portal. If users experience issues with this feature, check the mail server connection and its associated

external application connection are configured correctly and that public credentials are defined. See also Section 15.4, "Registering Mail Servers."

- For more information about setting up public credentials using Enterprise Manager, see Table 23–5, " External Application Connection - Shared User and Public User Credentials" in Chapter 23, "Managing External Applications."
- For more information about setting up public credentials using WLST, see the "addExtAppCredential" section in *Oracle WebLogic Scripting Tool*.

48.11.2 Enabling Anyone to Self-Register

When *anyone* is allowed to self-register (that is, any public user), a **Register** link or button displays on the WebCenter Portal login page (Figure 48–26).

Figure 48–26 Self-Registration Available on Login Page

New users must create an account before gaining access to WebCenter Portal.

Users who self-register are added directly to the WebCenter Portal identity store and assigned the `Authenticated-User` role. Out-of-the-box, users with `Authenticated-User` role have access to their own Home portal, pages that they create, and public pages. They are also allowed to view public portals, join any portal that allows self-subscription, and create portals of their own. If you enable self-registration, consider modifying `Authenticated-User` permissions to suit your exact requirements. See Section 43.4.4.2, "Modifying Application Role Permissions."

To allow anyone to self-register with WebCenter Portal:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **General**.

   You can also enter the following URL in your browser to navigate directly to the **General** page:

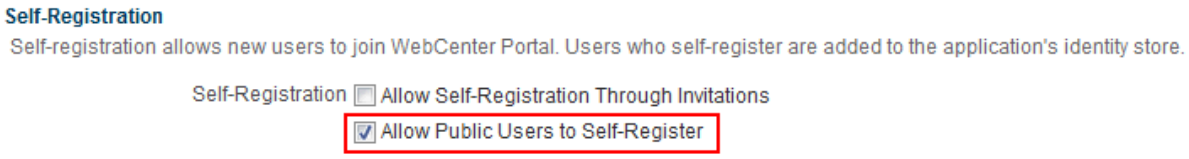
   `http://host:port/webcenter/portal/builder/administration/general`

   See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Select **Allow Public Users to Self-Register** (Figure 48–27).

   When you deselect this option, public users cannot self-register with WebCenter Portal. You still enable self-registration on an invitation-only basis if you want. See Section 48.11.3, "Enabling Self-Registration By Invitation-Only."

Figure 48–27 Allowing Public Users to Self Register



3. Click **Apply**.

See also the "Registering Yourself with WebCenter Portal" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

48.11.3 Enabling Self-Registration By Invitation-Only

Out-of-the-box, only registered WebCenter Portal users are candidates for portal membership. While this might meet the needs of most WebCenter Portal users, some portals will want to recruit members outside of the WebCenter Portal community.

The WebCenter Portal system administrator can extend portal membership to users outside of WebCenter Portal by allowing them to self-register on an *invitation-only* basis. When this facility is enabled, portal moderators can invite anyone to join their portal by sending them a customizable invitation by mail. The invitation includes a secure, self-registration URL which the invited party clicks to accept portal membership.

New members recruited in this way must create an account with WebCenter Portal before gaining access to the portal. Users who self-register by invitation are added to the identity store, and to the portal's member list.

Note: Users who self-register by invitation will also be assigned the default application role —`Authenticated-User`. Out-of-the box, users with the `Authenticated-User` role have access to their own Home portal, pages that they create, and public pages. They are also allowed to view public portals, join any portal that allows self-subscription, and create portals of their own. When you enable self-registration, consider modifying `Authenticated-User` permissions to suit your exact requirements. See also Section 43.4.4.2, "Modifying Application Role Permissions."

To allow anyone to self-register with WebCenter Portal through invitations:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **General**.

   You can also enter the following URL in your browser to navigate directly to the **General** page:

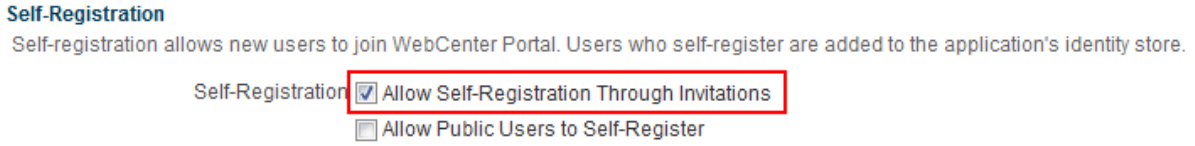
   `http://host:port/webcenter/portal/builder/administration/general`

See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Select **Allow Self-Registration Through Invitations** (Figure 48–28).

   When you deselect this option, only existing users are candidates for portal membership.

Figure 48–28 Allowing Self-Registration Through Invitations



3. Click **Apply**.

After you enable this option, portal moderators can invite anyone to become a member of their portal. See the "Inviting a Non–Registered User" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

48.12 Choosing a Default Look and Feel for New Pages

Administrators can set up a default look and feel for system, business role, and personal pages to simplify page creation for first-time users or to steer users toward a particular page scheme and style.

For more information:

- Chapter 50, "Customizing System Pages"
- Chapter 51, "Managing Business Role Pages"
- Chapter 52, "Managing Personal Pages"

Individuals may personalize the default settings in their Home portal view. For more information, see Section 52.2, "Setting Application-Level Page Creation Defaults for Personal Pages."

48.13 Enabling and Disabling Access to the Home Portal

Access to the Home portal is optional—it is not mandatory to provide users with a private work area where they can store personal content and perform personal tasks. Users can fully participate in collaborative projects without access to the Home portal.

Users who do not have access to the Home portal cannot use personal productivity tools (such as favorites), create personal pages, or see personal pages that other users might share.

The `Portal Server: View` permission controls whether users have access to the Home portal. Administrators can disable access to everyone using WebCenter Portal or to specific users only. Use Table 48–1 to determine which permission settings are required for the different roles.

To enable or disable access to the Home portal:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Security**.

You can also enter the following URL in your browser to navigate directly to the **Security** page:

`http://host:port/webcenter/portal/builder/administration/security`

> **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. On the **Roles** tab, under **Portal Server**, select or deselect the **View** check boxes for the roles, as appropriate (Table 48–1):

Table 48–1 Portal Server: View Permissions

Role	Select Portal Server: View	Deselect Portal Server: View
`Administrator`	Users assigned this role can always access the Home portal.	Not applicable for administrators.
`Application Specialist`	Users assigned this role can always access the Home portal.	Users with this role cannot access the Home portal. (Assumes the `Portal Server: View` permission is disabled for the Authenticated-User and the Public-User.)
`Public User`	Unauthenticated users can see personal pages and content marked public.	Unauthenticated users only see the login page.
`Authenticated User`	Everyone can access the Home portal.	Users cannot access the Home portal unless you grant them another role that specifies otherwise.
`Any Custom Role`	Users assigned the custom role have access to the Home portal.	Users with the custom role cannot access the Home portal. (Assumes the `Portal Server: View` permission is disabled for the Authenticated-User and the Public-User.)

3. Click **Apply** to save your changes.

   New permissions are effectively immediately.

48.14 Setting Up Defaults for WebCenter Portal Tools and Services

The system administrator is also responsible for setting up tools and services options for WebCenter Portal. For more information, see Chapter 8, "Managing Tools and Services."

49

Managing Security Across Portals

This chapter describes the tasks available on the **Security** page in WebCenter Portal Builder Administration. The system administrator can modify the default settings to suit the needs of the organization.

This chapter contains the following sections:

Permissions: To perform the tasks in this chapter, you must have the WebCenter Portal `Administrator` role or a custom role that grants the following permission:

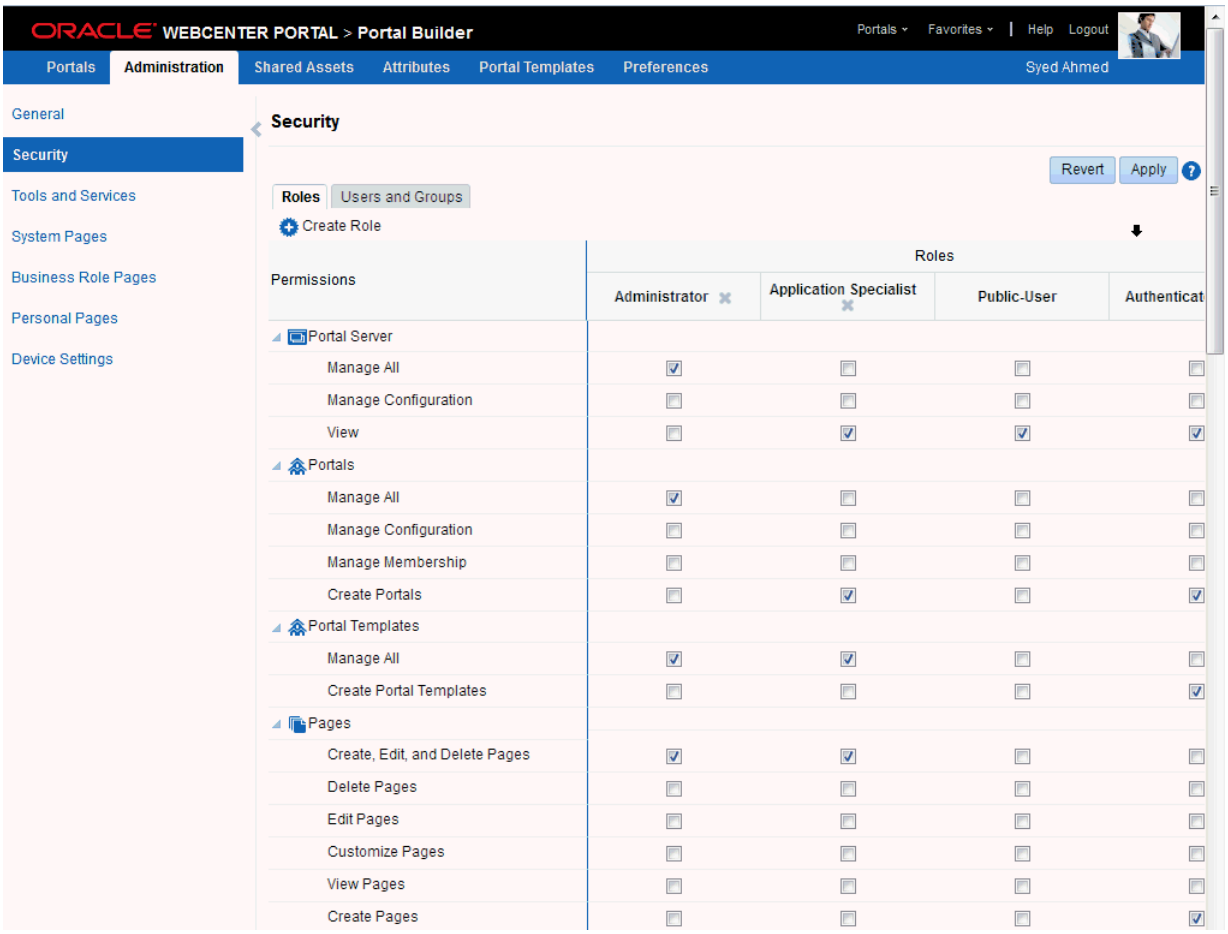
- `Portal Server-Manage Configuration`

Additionally, you need permissions to create and manage portals (`Portals-Create` and/or `Portals-Manage All`). For more information about permissions, see Section 49.3, "About Application Roles and Permissions."

49.1 About Portal Security

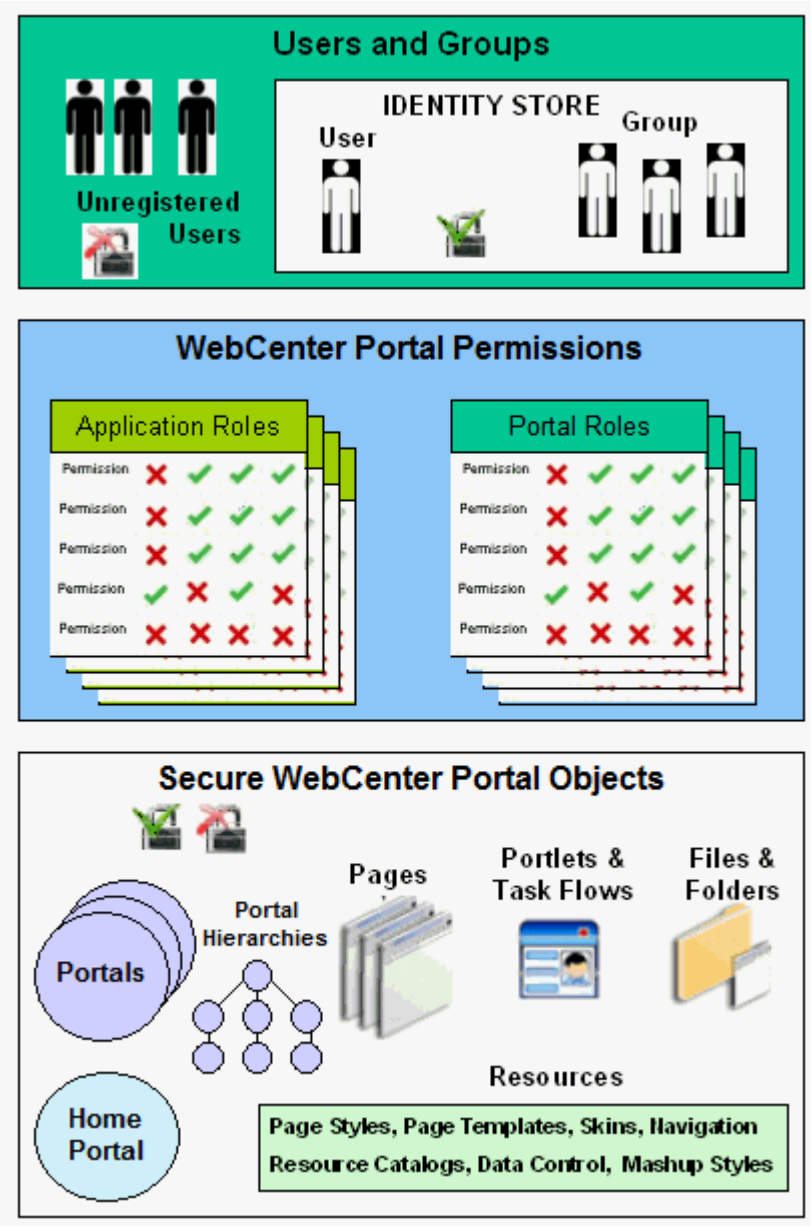
WebCenter Portal provides a comprehensive security model that enables you to control what users can see and change in WebCenter Portal. Using the **Security** page in WebCenter Portal Builder Administration (Figure 49–1), you can control which users (and groups) have access to individual portals and the Home portal and you can also control exactly what users and groups can see and do by enabling and disabling various permissions.

Figure 49–1 Portal Builder Administration: Security Page



Within a particular portal you can restrict user and group access to individual pages, page content (such as task flows, portlets, documents, and folders), and resources (such as page templates, page styles, skins, resource catalogs, and so on).

Figure 49–2 WebCenter Portal Security



User and Groups

A user is a single person in the identity store, and a group contains multiple users. In WebCenter Portal you can grant permissions to individual users and to groups of users.

Unregistered Users and Self-Registration

Self-registration allows unregistered users to create their own login and password for WebCenter Portal. A user who self-registers is immediately and automatically granted access to WebCenter Portal and a new user account is created in WebCenter Portal's identity store.

Application Roles and Portal Roles

Application roles determine what a user (or group) can see and do in the Home portal which, for some administrative functions, can impact all of WebCenter Portal. Portal roles control actions within a particular portal.

Portals and Portal Hierarchies

Portals support the formation and collaboration of project teams and communities of interest by providing a dedicated and readily accessible area for relevant services, pages, and content and by supporting the inclusion of specified members.

A portal hierarchy consists of a parent portal with one or more subportals. Subportals can inherit the security (members, roles, and permissions) of their parent.

Home Portal

The Home portal is a shared portal that, by default, is accessible to everyone who is logged in. Application roles apply while a user is working within the Home portal. In most applications, the Home portal focuses on social networking and personal content.

Resources

Various portal resources help define the overall structure, look and feel, and content in portals, and these include page templates, page styles, skins, navigation models, resource catalogs, content presenter display templates, task flow styles, data controls, and task flows. Users with appropriate privileges can build and customize resources for the entire application, a single portal, or a portal hierarchy.

Pages

Anyone authorized to edit a page can grant access and permissions to other users and groups. For example, you might grant view-only permission to everyone in the sales group, edit permission to sales managers, and manage permission to a single user. Alternatively, you can specify that the page inherits its access from the application.

Page Content, Files, and Folders

Some pages might contain content that you want only a select set of users, or even only one other user, to see. For example, a page aimed at sales people might include two Announcement task flows; one aimed at all sales people and the other at only sales managers. By restricting access to the second Announcement task flow, you can hide management-level announcements from anyone who is not a sales manager.

49.2 About Users

A WebCenter Portal user has a login account for WebCenter Portal—provisioned directly from an existing identity store. See also, Section 31.3, "Adding Users to the Embedded LDAP Identity Store."

All users in the identity store are assigned minimal privileges in WebCenter Portal through the `Authenticated-User` role. The only exception is the system administrator (`weblogic` by default); out-of-the-box, the system administrator is the only user assigned full administrative privileges through the `Administrator` role. For more information, read the next section Section 49.3.1.1, "Default Application Roles."

It is the system administrator's job to assign each user an appropriate application role. Alternatively, the system administrator may choose to assign the `Administrator` role to another user and delegate this responsibility.

Table 49–1 Default Administrator in WebCenter Portal

User	Description
System Administrator (`weblogic`)	Administrator for the entire application server, sometimes referred to as the super administrator or Fusion Middleware administrator. This user can manage any application on the server, including WebCenter Portal.

49.3 About Application Roles and Permissions

Application roles control the level of access a user has to information and services in WebCenter Portal. Specifically, application roles and their permissions determine what a user can see and do in their *Home portal*.

This section includes:

About Application Roles

About Application Permissions

49.3.1 About Application Roles

Application role assignment is the responsibility of the WebCenter Portal administrator. Administrators can assign users a default application role or create additional, custom roles specific to their WebCenter Portal deployment. For more detail, see:

- Default Application Roles
- Custom Application Roles

Application roles apply when users are working within their Home portals. A different set of roles and permissions apply when a user is working within a particular portal. It is the portal moderator's responsibility to determine suitable role assignments for each of its members. See also, Section 49.6, "Managing Application Roles and Permissions," and the "Administering Security in a Portal" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

> **Note:** Application roles and permissions defined within WebCenter Portal are stored in its *policy store* and, consequently, apply to this WebCenter Portal only. Enterprise roles are different; enterprise roles are stored within the application's *identity store* and do not imply any permissions within WebCenter Portal. See Section 30.2.2, "Application Roles and Enterprise Roles."

49.3.1.1 Default Application Roles

WebCenter Portal provides several default application roles (Table 49–2). You cannot delete default application roles but you can modify the default permission assignments for each role. For more information, see Section 49.6.2, "Modifying Application Role Permissions."

Table 49–2 Default Application Roles for WebCenter Portal

Application Role	Description	Modify?
Administrator	Users with the `Administrator` role can set application-wide properties for WebCenter Portal, create business role pages, configure defaults for discussion forums, mail, and people connection services, register producers and external applications, as well as perform other administrative duties such as editing the login page and the self-registration page. Administrators can also manage users and roles for the WebCenter Portal, delegate or revoke privileges to/from other users, manage portals and portal templates, and also import and export portal information. Out-of-the-box, the system administrator is the only user assigned full administrative privileges for the WebCenter Portal through the `Administrator` role.	Yes* *Except for Application permissions which are read-only
Application Specialist	Users with the `Application Specialist` role can create portals; manage portal templates; create, edit, and delete pages, page styles, page templates, Content Presenter templates, data controls, navigations, pagelets, resource catalogs, skins, task flow styles, and task flows; update People Connections data, and connect with people.	Yes
Authenticated-User	Authenticated users of WebCenter Portal are granted the `Authenticated-User` role. Users who login are assigned with this role and, by default, have access to their own Home portal, pages that they create, and public pages. These users can also view public portals, create portals, and create portal templates. This role inherits permissions from the `Public-User` role. In the WebCenter Portal, the `Authenticated-User` role is equivalent to `authenticated-role`—a standard OPSS (Oracle Platform Security Services) role.	Yes
Public-User	Anyone with access to the WebCenter Portal who is not logged in, is granted the `Public-User` role. Such users are anonymous, unidentified, and can see public content only. In the WebCenter Portal, the `Public-User` role is equivalent to `anonymous-role`—a standard OPSS (Oracle Platform Security Services) role.	Yes

49.3.1.2 Custom Application Roles

Custom application roles (sometimes known as user-defined roles) are specific to your WebCenter Portal. When setting up WebCenter Portal, it is the WebCenter Portal administrator's job to identify which application roles are required, select suitable role names, and define the responsibilities of each role.

For example, an education environment might require roles such as Teacher, Student, and Guest. While roles such as Finance, Sales, Human Resources, and Support would be more appropriate for a corporate environment.

In WebCenter Portal, custom application roles inherit permissions from the `Authenticated-User` role.

To learn how to set up application roles for WebCenter Portal users, see

49.3.2 About Application Permissions

Every application role has specific, defined capabilities known as permissions. These permissions allow individuals to perform specific actions in the Home portal.

Permissions are categorized as follows and listed individually in the subsequent tables:

- Portal Server
- Portals
- Portal Templates
- Pages
- Content Presenter Templates
- Data Controls
- Discussions
- Links
- Navigations
- Page Styles
- Page Templates
- Pagelets
- People Connections
- Resource Catalogs
- Skins
- Task Flow Styles
- Task Flows

No permission, except for `Manage All`, inherits privileges from other permissions.

Table 49–3 Application Permissions in WebCenter Portal

Category	Application Permissions
Portal Server	**Manage All** - Enables access to all *WebCenter Portal Administration* pages: Portals, Administration, Shared Assets, Attributes, Portal Templates, and Preferences. Through these pages, users can manage application security (users/roles), configure application-wide properties and services, manage resources, create business role pages, manage everyone's personal pages, customize system pages, view portals accessible to them, as well as export/import portals and portal templates. Some administrative tasks are exclusive to the out-of-the-box `Administrator` role and cannot be performed by granting the `Application-Manage All` permission. These tasks include editing the login page, the self-registration page, and profile gallery pages, as well as the ability to manage *all* portals, *all* portal templates, external applications, and portlet producers. **Manage Configuration** - Same as the `Application-Manage All` permission but excludes security privileges. Users with this permission cannot access the Administration - Security page. **View** - Enables users to view WebCenter Portal, and gives user access to the Home portal. See also Section 49.6.3, "Granting Permissions to the Public-User," and Section 49.6.4, "Granting Permissions to the Authenticated-User."
Portals	**Manage All** - Enables access to all portal administration pages (Overview, Settings, Pages, Assets, Attributes, Security, Tools and Services, Subportals, System Pages). Through these pages users can manage portal membership, assign permissions and roles, manage, delete, and export portals and resources, set portal properties, and manage service availability. **Manage Configuration** - Same as the `Manage All` permission but excludes security privileges. Users with this permission cannot access the Security pages unless they are a portal moderator. **Manage Membership** - Enables users to manage portal membership through Security pages. **Create Portals** - Enables users to create portals.
Portal Templates	**Manage All** - Enables users to manage any portal template (through the Portal Templates page) and delete templates accessible to them. See also, the "Managing All Portal Templates" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*. **Create Portal Templates** - Enables users to create portal templates.
Pages	**Create, Edit, and Delete Pages** - Enables users to create, edit and delete pages in their Home portal. **Delete Pages** - Enables users to delete pages in their Home portal. **Edit Pages** -Enables users to add or edit personal page content, rearrange content, and set page parameters and properties. **Customize Pages** - Enables users to customize their view of pages in the Home portal by adding, editing, or removing content. **View Pages** - Enables users to view pages in the Home portal. **Create Pages** - Enables users to create or design a new page for their Home portal. These permissions only apply to the Home portal. The permissions do not apply to pages that are created within a portal. Page permissions within a portal are granted on a per-portal basis by the portal moderator. See the "Managing Roles and Permissions for a Portal" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Table 49–3 (Cont.) Application Permissions in WebCenter Portal

Category	Application Permissions
Content Presenter Templates	**Create, Edit, and Delete Content Presenter Templates** - Enables users to create, edit and delete content display templates for the application through Portal Builder. **Create Content Presenter Templates** - Enables users to create content display templates for the application. **Edit Content Presenter Templates** - Enables users to edit application-level content display templates. See also, the "Publishing Content Using Content Presenter" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.
Data Controls	**Create, Edit, and Delete Data Controls** - Enables users to create, edit and delete data controls for the application through Portal Builder. **Create Data Controls** - Enables users to create data controls for the application. **Edit Data Controls** - Enables users to edit application-level data controls. See also, the "Working with Data Controls" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.
Discussions	**Create, Edit, and Delete Discussions** - Enables users to manage categories, forums, and topics on the back-end discussions server and set discussion forum properties for all portals. See also, "Understanding Discussion Server Role Mapping"
Links	**Create and Delete Links** - Enables users to create and delete links between objects, and manage link permissions. **Create Links** - Enables users to create links between objects, and delete links that they create. **Delete Links** - Enables users to delete a link between two objects.
Navigations	**Create, Edit, and Delete Navigations** - Enables users to create, edit, and delete navigations for the application through Portal Builder. **Create Navigations** - Enables users to create navigations for the application. **Edit Navigations** - Enables users to edit application-level navigations. See also, the "Working with Portal Navigation" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.
Page Styles	**Create, Edit, and Delete Page Styles** - Enables users to create, edit, and delete page styles through Portal Builder. **Create Page Styles** - Enables users to create page styles for the application. **Edit Page Styles** - Enables users to edit application-level page styles. See also, the "Working with Page Styles" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.
Page Templates	**Create, Edit, and Delete Page Templates** - Enables users to create, edit, and delete page templates through Portal Builder. **Create Page Templates** - Enables users to create page templates for the application. **Edit Page Templates** - Enables users to edit application-level page templates. See also, the "Working with Page Templates" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.
Pagelets	**Create, Edit, and Delete Pagelets** - Enables users to create, edit, and delete pagelets through Portal Builder. **Create Pagelets** - Enables users to create pagelets for the application. **Edit Pagelets** - Enables users to edit application-level pagelets. See also, the "Working with Pagelets" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Table 49–3 (Cont.) Application Permissions in WebCenter Portal

Category	Application Permissions
People Connections	**Manage People Connections** - Enables users to manage application-wide settings for People Connection services. **Update People Connections Data** - Enables users to edit content associated with People Connection services. **Connect with People** - Enables users to share content associated with People Connection services with others.
Resource Catalogs	**Create, Edit, and Delete Resource Catalogs** - Enables users to create, edit and delete resource catalogs for the application through Portal Builder. **Create Resource Catalogs** - Enables users to create resource catalogs for the application. **Edit Resource Catalogs** - Enables users to edit application-level resource catalogs. See also, the "Working with Resource Catalogs" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.
Skins	**Create, Edit, and Delete Skins** - Enables users to create, edit, and delete skins through Portal Builder. **Create Skins** - Enables users to create skins for the application. **Edit Skins** - Enables users to edit application-level skins. See also, the "Working with Skins" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.
Task Flow Styles	**Create, Edit, and Delete Task Flow Styles** - Enables users to create, edit, and delete content display templates for the application through Portal Builder. **Create Task Flow Styles** - Enables users to create content display templates for the application. **Edit Task Flow Styles** - Enables users to edit application-level content display templates. See also, the "Publishing Content Using Content Presenter" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.
Task Flows	**Create, Edit, and Delete Task Flows** - Enables users to create, edit, and delete task flows based on a task flow style through Portal Builder. **Create Task Flows** - Enables users to create task flows for the application. **Edit Task Flows** - Enables users to edit application-level task flows. See also, the "Working with Task Flows" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

49.3.2.1 Understanding the Default Permissions

Table 49–4 shows the default permissions assigned to out-of-the-box application roles.

✔ - Shows an explicitly granted permission or action.

✙ - Shows an implied permission because of an explicitly granted permission.

Table 49–4 Default Application Roles and Permissions in WebCenter Portal

	Default Application Roles			
Permissions	**Administrator**	**Application Specialist**	**Authenticated -User**	**Public-User**
Portal Server				
Manage All	✔			
Manage Configuration	✙			
View	✙	✔	✔	✔

Table 49–4 (Cont.) Default Application Roles and Permissions in WebCenter Portal

	Default Application Roles			
Permissions	**Administrator**	**Application Specialist**	**Authenticated -User**	**Public-User**
Portals				
Manage All	✔			
Manage Configuration				
Manage Membership				
Create Portals		✔	✔	
Portal Templates				
Manage All	✔	✔		
Create Portal Templates			✔	
Pages				
Create, Edit, and Delete	✔	✔		
Delete				
Edit				
Customize				
View				
Create			✔	
Content Presenter Templates				
Create, Edit, and Delete	✔	✔		
Create				
Edit				
Data Controls				
Create, Edit, and Delete	✔	✔		
Create				
Edit				
Discussions				
Create, Edit, and Delete	✔			
Links				
Create and Delete	✔			
Create				
Delete				
Task Flow Styles				
Create, Edit, and Delete	✔	✔		
Create				
Edit				
Navigations				
Create, Edit, and Delete	✔	✔		

Table 49–4 (Cont.) Default Application Roles and Permissions in WebCenter Portal

	Default Application Roles			
Permissions	**Administrator**	**Application Specialist**	**Authenticated -User**	**Public-User**
Create				
Edit				
Page Styles				
Create, Edit, and Delete	✔	✔		
Create				
Edit				
Page Templates				
Create, Edit, and Delete	✔	✔		
Create				
Edit				
Pagelets				
Create, Edit, and Delete	✔	✔		
Create				
Edit				
People Connections				
Manage People Connections	✔			
Update People Connections Data		✔	✔	
Connect with People		✔	✔	
Resource Catalogs				
Create, Edit, and Delete	✔	✔		
Create				
Edit				
Skins				
Create, Edit, and Delete	✔	✔		
Create				
Edit				
Task Flows				
Create, Edit, and Delete	✔	✔		
Create				
Edit				

49.3.2.2 Understanding Discussion Server Role Mapping

Some WebCenter Portal services that need access to remote (back-end) resources also require role-mapping based authorization, that is, the WebCenter Portal roles that allow users to work with the Discussions service in WebCenter Portal, must be mapped to corresponding roles on WebCenter Portal's discussions server.

WebCenter Portal uses *application roles* to manage user permissions in the Home portal and *portal roles* to manage user permissions within a particular portal. On WebCenter Portal's discussions server, a different set of roles and permissions apply.

Users who are working with discussions and announcements in WebCenter Portal automatically map to the appropriate discussions server role, shown in Table 49–5 and Table 49–6.

Table 49–5 Discussions Server Roles and Permissions - Application

Discussion Server Role	Discussion Server Permissions	WebCenter Portal Equivalent Application Permission
Administrator	Category Admin	`Discussions-Create, Edit, and Delete` Create, read, update and delete sub categories, forums, and topics inside the category for which permissions are granted.

Table 49–6 Discussions Server Roles and Permissions - For a Portal

Discussion Server Role	Discussion Server Permissions	WebCenter Portal Equivalent Permissions in a Portal
Moderator	Category Admin Forum Admin	▪ `Discussions-Create, Edit, and Delete` Create, read, update and delete forums and topics. ▪ `Announcements-Create, Edit, and Delete` Create, read, update and delete announcements.
	Create Message Create Announcement	▪ `Discussions-Create and Edit` Create and edit topics. ▪ `Announcements-Create and Edit` Create and edit announcements.
	Read Forum Create Thread	▪ `Discussions-Reply To` Reply to discussion topics.
	Read Forum	▪ `Discussions-View` View forums and topics. ▪ `Announcements-View` View announcements.

Any user assigned the `Application-Discussions-Create Edit Delete` permission in WebCenter Portal is automatically added to WebCenter Portal's discussions server and assigned the `Administrator` role with the `Category Admin` permission. Out-of-the box, WebCenter Portal assigns the `Application-Discussions-Create Edit Delete` permission to the `Administrator` role only.

Similarly, in a given portal, any member assigned discussion and announcement permissions is granted the corresponding permissions on the discussions server.

49.3.2.3 Understanding Enterprise Group Role Mapping

In WebCenter Portal you can assign individual users or multiple users in the same enterprise group to WebCenter Portal roles. Subsequent enterprise group updates in the back-end identity store are automatically reflected in WebCenter Portal. Initially, when you assign an enterprise group to a WebCenter Portal role, everyone in the

enterprise group is granted that role. If someone moves out of the group, the role is revoked. If someone joins the group, they are granted the role

For WebCenter Portal to properly maintain enterprise group-to-role mappings, back-end servers, such as the discussions server and content server, must support enterprise groups too. WebCenter Portal's Discussion Server and WebCenter Content's Content Server versions provided with this release both support enterprise groups but previous versions may not. See also, Section 49.7, "Troubleshooting Issues with Users and Roles."

49.4 About Roles and Permissions within a Portal

When a user becomes a member of a particular portal, a different set of roles and responsibilities apply. For details, see the "Administering Security in a Portal" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

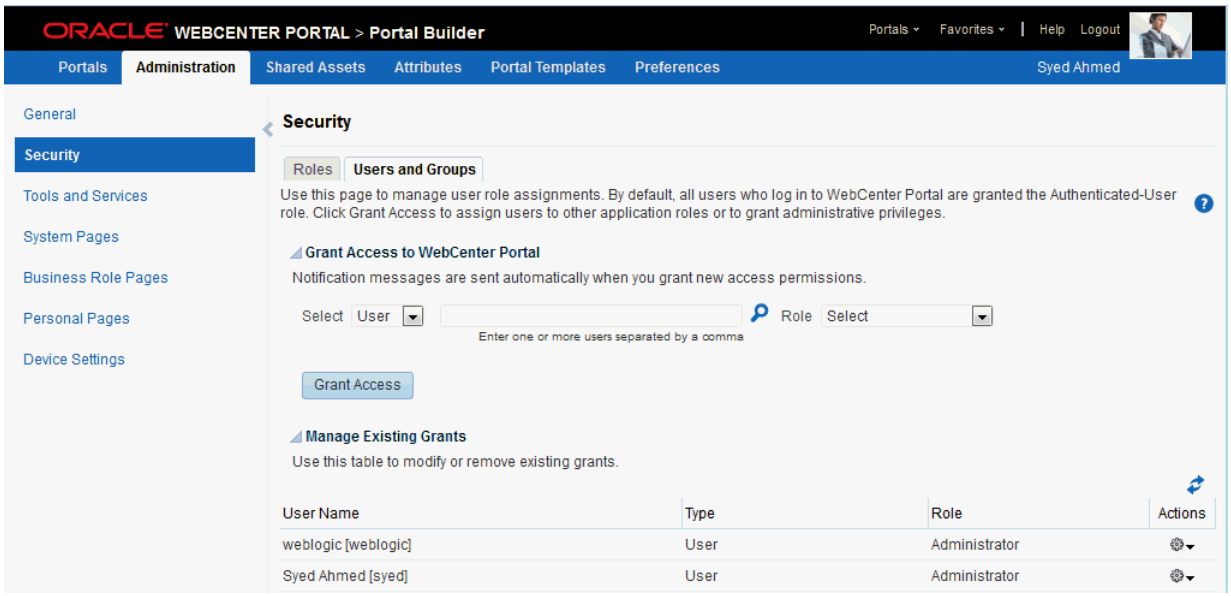
49.5 Managing Users

Administrators must ensure that all WebCenter Portal users have appropriate permissions. To get permissions, users must be assigned to an appropriate application role.

From the **Users and Groups** page (Figure 49–3), administrators can manage application roles for all the users who have access to WebCenter Portal, that is, all users defined in the identity store. From here, you can change user role assignments, grant administrative privileges, and revoke user permissions. To access the Users and Groups page, open Portal Administration and then click the **Security** page. For details, see the "Accessing Portal Administration" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Only users granted special (non-default) application privileges appear in this table. Initially, all users in the WebCenter Portal identity store are assigned minimal privileges through the `Authenticated-User` role. Users with the default `Authenticated-User` role are not listed here. See also Section 49.3.1.1, "Default Application Roles."

Figure 49–3 WebCenter Portal Administration - Users and Groups Page



This section tells you how to assign roles and contains the following subsections:

- Section 49.5.1, "Assigning Users (and Groups) to Roles"
- Section 49.5.2, "Assigning a User to a Different Role"
- Section 49.5.3, "Giving a User Administrative Privileges"
- Section 49.5.4, "Revoking Application Roles"
- Section 49.5.5, "Adding or Removing Users"

49.5.1 Assigning Users (and Groups) to Roles

Initially, all users in the WebCenter Portal identity store are assigned minimal privileges through the `Authenticated-User` role. You can assign individual users (or multiple users in the same enterprise group) to a different application role through WebCenter Portal Administration.

Updates in your back-end identity store, such as new users or someone leaving an enterprise group, are automatically reflected in WebCenter Portal. Initially, when you assign an enterprise group to a WebCenter Portal role, everyone in the enterprise group is granted that role. If someone moves out of the group, the role is revoked. If someone joins the group, they are granted the role.

> **Note:** For WebCenter Portal to properly maintain enterprise group-to-role mappings, back-end servers, such as the discussions server and content server, must support enterprise groups too. When back-end servers do not support enterprise groups, the message "`Group [name] not found in the Identity Store`" displays. See also Section 49.7, "Troubleshooting Issues with Users and Roles."

To assign a user (or a group of users) to a different application role:

1. Open WebCenter Portal Administration.

For details, see Section 47.2, "Accessing the Portal Builder Administration Page."

2. Click **Security**, then **Users and Groups** (Figure 49–3).

   This page lists users to which additional roles are defined.

3. Choose **User** or **Group** from the drop-down list.

   Select **User** to grant permissions to one or more users defined in the identity store. Select **Group** to grant permissions to groups of users.

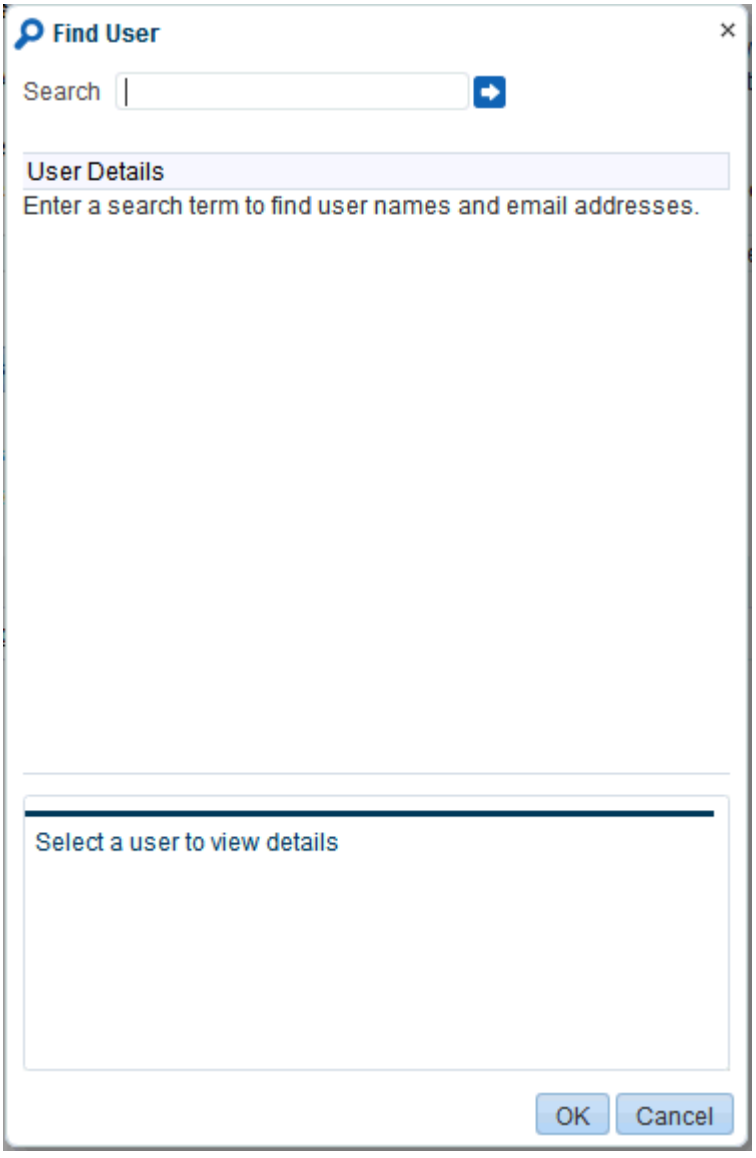
4. If you know the exact name of the user or group, enter the name in the text box, separating multiple names with commas.

   If you are not sure of the name you can search your identity store:

   a. Click the **Find** icon ( ).

      The Find User (or Find Group) dialog box opens (Figure 49–4).

Figure 49–4 Finding Users and Groups in the Identity Store



b. Enter a search term for a user or group, then click the **Search** icon.

For tips on searching for a user or group in the identity store, see the "Searching for a User or Group in the Identity Store" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Users (or groups) matching your search criteria display in the **Select User** dialog box. For more details on which fields are searched, see the "Searching for a User or Group in the Identity Store" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*

Tip: ■

- Use * as a wildcard, for example `*sales`.
- Leave the search field blank to list all users (or groups) in the identity store.
- Enter a space between two search terms to search First Name and Last Name, for example `jo sm`, searches for `jo` in First Name and `sm` in Last Name.

c. Select one or more names from the list.

To assign roles to multiple users or groups, multi-select all the names required. **Ctrl-Click** rows to select multiple names.

d. Click **OK**.

The names that you select appear on the **User and Groups** tab.

5. To assign a role, select a **Role** from the drop-down list.

Select an appropriate role for the selected users (or groups). Only choose **Administrator** to assign full, administrative privileges for WebCenter Portal.

If the role you want is not listed, create a new role that meets your requirements (see Section 49.6.1, "Defining Application Roles").

When no role is selected, the user assumes the `Authenticated-User` role. See Section 49.3.1.1, "Default Application Roles."

6. Click **Grant Access**.

User/user group names and new role assignment appear in the table.

Note: Group names are clickable, enabling you to drill down to see user names of the current group members.

A list of members does not display for a dynamic group based on Oracle Entitlements Server (OES) roles since OES roles are based on dynamic attributes and therefore do not have any static members. See also Section 31.8, "Configuring Dynamic Groups for WebCenter Portal."

49.5.2 Assigning a User to a Different Role

From time to time, a user's role in WebCenter Portal may change. For example, a user may move out of sales into the finance department and in this instance, the user's role assignment may change from *Sales* to *Finance*.

Note: You cannot modify your own role or the system administrator's role. See Section 49.3.1, "About Application Roles."

To assign a user to a different role:

1. Open WebCenter Portal Administration.

   For details, see Section 47.2, "Accessing the Portal Builder Administration Page."

2. Click **Security**, then **Users and Groups** (Figure 49–3).

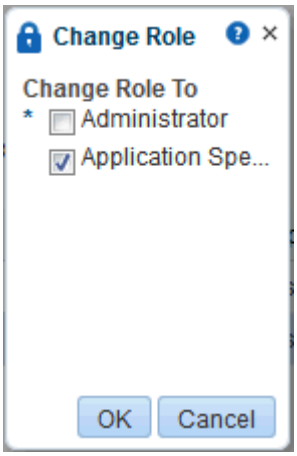
3. In the **Manage Existing Grants** table, scroll down to the user you want.

   Only users with non-default role assignments are listed in the table. If the user you want is not listed, grant the role required as described in Section 49.5.1, "Assigning Users (and Groups) to Roles."

4. Click the **Actions** icon, then choose **Change Role** from the drop-down list.

   The Change Role dialog box opens (Figure 49–5).

Figure 49–5 Changing a User's Application Role



5. Select roles as follows:

   - Select **Administrator** to assign full, administrative privileges for WebCenter Portal.
   - Select one or more roles from the list available.

     If the role you want is not listed, create a new role that meets your requirements (see Section 49.6.1, "Defining Application Roles").

     At least one role must be selected. To revoke all role assignments, reverting user permissions to the default `Authenticated-User` role, see Section 49.5.4, "Revoking Application Roles".

6. Click **OK**.

New role assignments display in the table.

49.5.3 Giving a User Administrative Privileges

It is easy to give a user full, administrative privileges for WebCenter Portal through the `Administrator` role. Administrators have the highest privilege level and can view

and modify anything in WebCenter Portal so take care when assigning the `Administrator` role.

Some administrative tasks are exclusive to the `Administrator` role and cannot be performed by granting the `Application-Manage All` permission. These tasks include editing the login page, the self-registration page, and profile gallery pages. See also Section 49.3.1.1, "Default Application Roles."

To give a user administrative privileges:

1. Open WebCenter Portal Administration.

   For details, see Section 47.2, "Accessing the Portal Builder Administration Page."

2. Click **Security**, then **Users and Groups** (Figure 49–3).

   The Role column indicates which users already have full administrative privileges through the `Administrator` role.

3. In the **Manage Existing Grants** table, scroll down to the user you want.

   Only users with non-default role assignments are listed in the table. If the user you want is not listed, follow steps in Section 49.5.1, "Assigning Users (and Groups) to Roles" to grant the `Administrator` role.

4. Click the **Actions** icon, then choose **Change Role** from the drop down list.

   The Change Role dialog box opens (Figure 49–5).

5. Select **Administrator** to assign full, administrative privileges for WebCenter Portal.

6. Click **OK**.

The new role assignment displays in the table.

49.5.4 Revoking Application Roles

It is easy to revoke application role assignments that no longer apply. You can revoke roles individually or revoke all application roles assigned to a particular user at once.

Revoking all of a user's application roles does not remove that user from the identity store and the user still has access to WebCenter Portal through the default `Authenticated-User` role.

> **Note:** You cannot revoke your own role assignments or the system administrator's role. See Section 49.3.1, "About Application Roles."

To revoke application roles:

1. Open WebCenter Portal Administration.

   For details, see Section 47.2, "Accessing the Portal Builder Administration Page."

2. Click **Security**, then **Users and Groups** (Figure 49–3).

   This page lists users to which additional roles are defined.

3. In the **Manage Existing Grants** table, scroll down to the user you want.

4. Click the **Actions** icon:

   - Choose **Change Role** icon to revoke one or more, specific application roles. See also Section 49.5.2, "Assigning a User to a Different Role".

- Choose **Delete Role Assignments** to revoke all roles assigned to that user, and then click **Delete** when asked for confirmation.

Access for that user is revoked immediately.

When you delete all the roles assigned to a particular user, the user is no longer listed on the Users and Groups page. The user remains in the identity store and still has access to WebCenter Portal through the `Authenticated-User` role. See Section 49.3.1.1, "Default Application Roles."

49.5.5 Adding or Removing Users

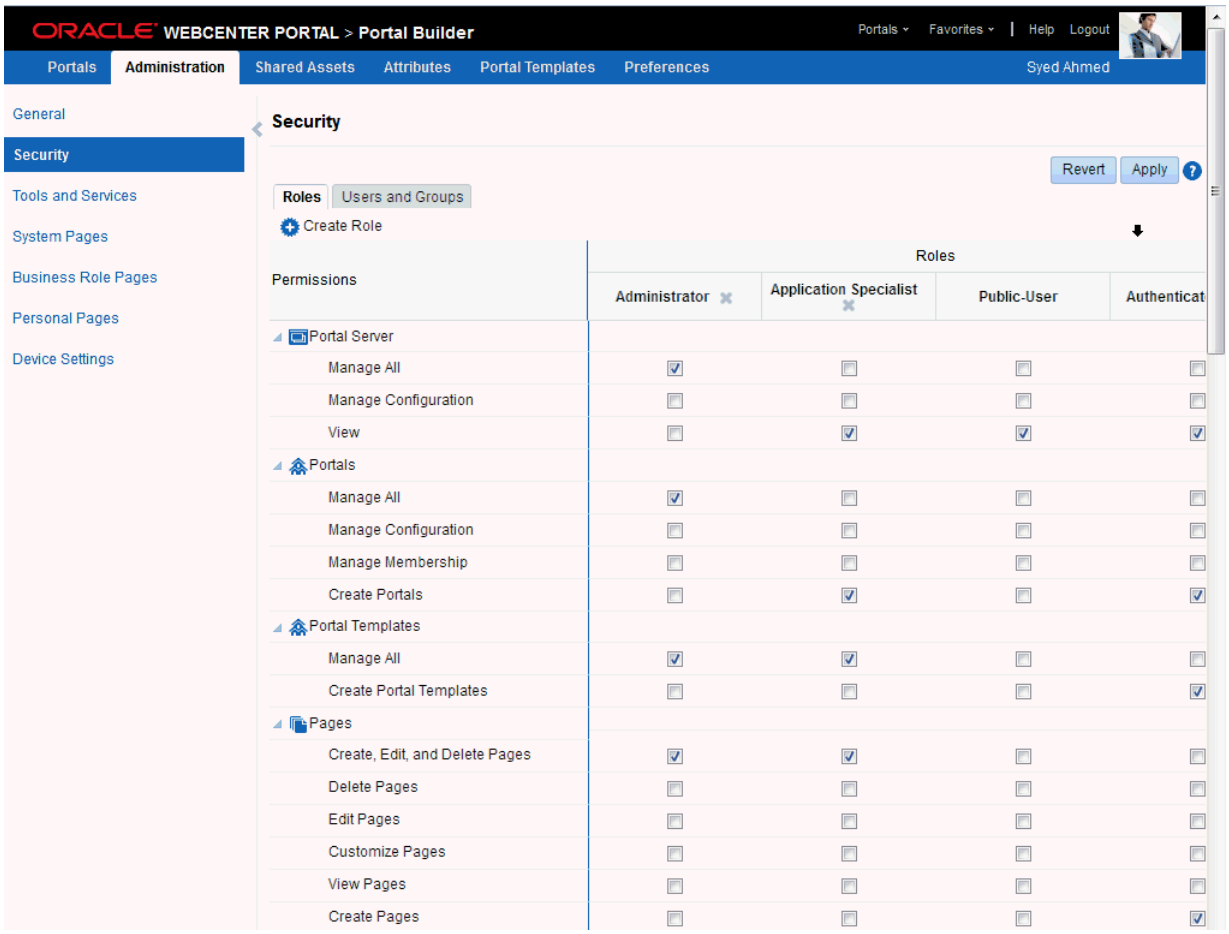
WebCenter Portal administrators cannot add new user data directly to the WebCenter Portal identity store or remove user credentials. Identity store management is the responsibility of the systems administrator and takes place through the WLS Administration Console or directly into embedded LDAP identity stores using LDAP commands. See also Section 31.3.1, "Adding Users to the Identity Store Using the WLS Administration Console."

WebCenter Portal administrators can, however, enable self-registration for the application. Through self-registration, invited and uninvited users can create their own login and password for WebCenter Portal. A user who self registers is immediately and automatically granted access to WebCenter Portal and a new user account is created in the identity store. See also Section 48.11, "Enabling Self-Registration."

49.6 Managing Application Roles and Permissions

WebCenter Portal uses application roles to manage permissions for users working in their *Home portal*. Administrators manage application roles and permissions on the Roles page (Figure 49–6).

Figure 49–6 WebCenter Portal Administration - Roles Page



This section tells you how to manage application roles, and their permissions from WebCenter Portal Administration pages. It contains the following subsections:

- Section 49.6.1, "Defining Application Roles"
- Section 49.6.2, "Modifying Application Role Permissions"
- Section 49.6.3, "Granting Permissions to the Public-User"
- Section 49.6.4, "Granting Permissions to the Authenticated-User"
- Section 49.6.5, "Deleting Application Roles"

49.6.1 Defining Application Roles

Use roles to characterize groups of WebCenter Portal users and determine what they can see and do in their Home portals.

When defining application roles, use self-descriptive role names and try to keep the role policy as simple as possible. Choose as few roles as you can, while maintaining an effective policy.

Take care to assign appropriate access rights when assigning permissions for new roles. Do not allow users to perform more actions than are necessary for the role but at the same time, try not to inadvertently restrict them from activities they must perform. In some cases, users might fall into multiple roles.

To define a new application role:

1. Open WebCenter Portal Administration.

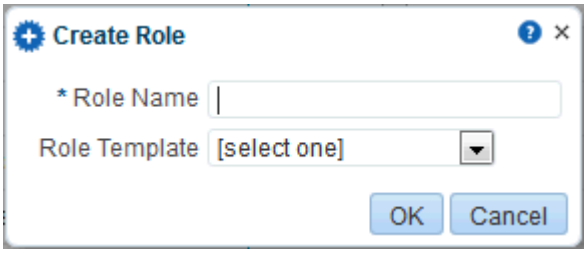
   For details, see Section 47.2, "Accessing the Portal Builder Administration Page."

2. Click **Security**, then **Roles** (Figure 49–6).

   Current application roles for WebCenter Portal display as columns in the table.

3. Click **Create Role** to define a new role for WebCenter Portal users.

Figure 49–7 Creating a New Role



4. Enter a suitable name for the role.

   Ensure the role names are self-descriptive. Make it as obvious as possible which users should belong to which roles. Role names can contain alphanumeric characters, blank spaces, @, and underscores.

5. (Optional) Choose a **Role Template**.

   The new role inherits permissions from the role template. You can modify these permissions in the next step.

   Choose **Administrator** to create a role that inherits full, administrative privileges. Conversely, choose **Public-User** to create a role that *typically* provides minimal privileges. Alternatively, choose a custom application role to be your template.

6. Click **OK**.

   The new role appears as a column in the table. The permissions list shows which actions users with this role can perform.

7. To modify user permissions for the role, select or clear each permission check box.

8. Click **Apply** to save any changes that you make to the role's permissions.

49.6.2 Modifying Application Role Permissions

Administrators can modify the permissions associated with application roles at any time. Application permissions are described in Section 49.3.2, "About Application Permissions."

Application role permissions allow individuals to perform specific actions in their Home portal. No permission, except for `Manage All`, inherits privileges from other permissions.

> **Note:** Application permissions cannot be modified for the `Administrator` role. See also Section 49.3.1.1, "Default Application Roles."

To change the permissions assigned to a role:

1. Open WebCenter Portal Administration.

   For details, see Section 47.2, "Accessing the Portal Builder Administration Page."

2. Click **Security**, then **Roles** (Figure 49–6).

   Current application roles for WebCenter Portal display as columns in the table.

3. Select or deselect **Permissions** check boxes to enable or disable permissions for a role.

4. Click **Apply** to save.

The new permissions are effective immediately.

49.6.3 Granting Permissions to the Public-User

Anyone who is not logged in to WebCenter Portal assumes the `Public-User` role. Out-of-the-box, the `Public-User` role is granted minimal privileges, that is, only the `View Application` permission.

> **Caution:** Take care when granting permissions to the `Public-User` role. Avoid granting administrative permissions such as `Application-Manage All`, `Application-Manage Configuration`, or any permission that might be considered unnecessary. See also Section 49.3.2, "About Application Permissions."

Granting the Application-View Permission

The `View Application` permission allows unauthenticated users to see public WebCenter Portal pages, such as the Welcome page, and also content that individual users choose to make public.

When `View Application` permission is granted to the `Public-User` role:

- Ensure that users understand that any personal page or personal content they choose to make public will become accessible to unauthenticated users outside of the WebCenter Portal community, that is, anyone with Web access.
- Consider customizing the default Welcome page that displays to public users before they login (Welcome Page). See Chapter 50, "Customizing System Pages."

If you do not want unauthenticated users to see WebCenter Portal content that is marked 'public', do not grant the `View Application` permission to the `Public-User` role. When public access is disabled, public content cannot be seen by unauthenticated users. Also, the Welcome page for WebCenter Portal is not displayed; public users are directed straight to a login page. Administrators may customize the default login page, if required. See Section 50.2, "Customizing System Pages for All Portals."

Granting Other Permissions

Be careful when assigning permissions to the `Public-User` role. For security reasons, Oracle recommends that you limit what anonymous users can see and do in WebCenter Portal.

49.6.4 Granting Permissions to the Authenticated-User

Anyone who is logged in to WebCenter Portal assumes the `Authenticated-User` role. Out-of-the-box, the `Authenticated-User` role is granted minimal privileges, through the following permissions: `View Application`, `Portals-Create`, `Portal`

`Templates-Create`, `Pages-Create`, `Update People Connections Data`, and `Connect with People`.

Other important notes:

- The `Authenticated-User` role always inherits permissions from the `Public-User` role.
- Custom application roles all inherit permissions from the `Authenticated-User` role.

49.6.5 Deleting Application Roles

When an application role is no longer required you should remove it from WebCenter Portal. This helps maintain a valid role list, and prevents inappropriate role assignment.

Application roles are deleted even when users are still assigned to the them. As you cannot delete any default roles, WebCenter Portal users will always have the `Authenticated-User` role.

> **Note:** Default roles cannot be deleted (`Administrator`, `Authenticated-User`, `Public-User`). See Section 49.3.1.1, "Default Application Roles."

To delete an application role:

1. Open WebCenter Portal Administration.

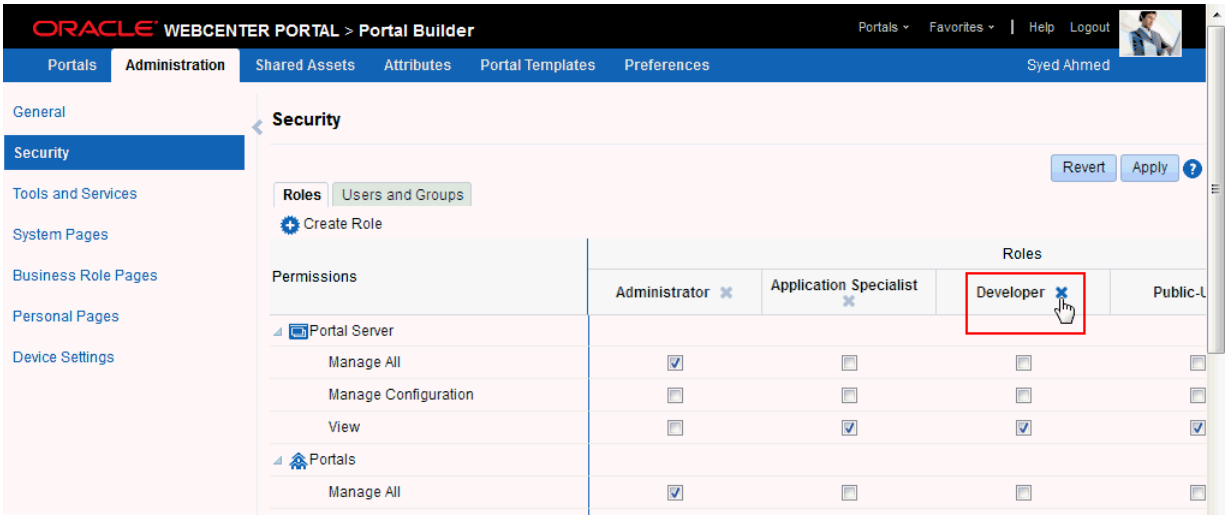
   For details, see Section 47.2, "Accessing the Portal Builder Administration Page."

2. Click **Security**, then **Roles** (Figure 49–6).

   Current application roles for WebCenter Portal display as columns in the table.

3. Select the **Delete Role** icon next to the role you want to delete (Figure 49–8).

Figure 49–8 Deleting an Application Role



4. Click **Delete** to confirm that you want to delete the role.

The role is removed from the table. Any users assigned to this role only, assume the default `Authenticated-User` role and do not appear on the Users and Groups tab.

49.7 Troubleshooting Issues with Users and Roles

For WebCenter Portal to properly maintain enterprise group-to-role mappings, the back-end discussions server and content server must support enterprise groups. The WebCenter Portal's Discussion Server and WebCenter Content's Content Server versions provided with Oracle WebCenter Portal 11.1.1.2.0 and later both support enterprise groups but previous versions may not.

If a back-end server *does not* support enterprise groups, an error message similar to the following displays when you try to add a group.

```
Warning: Group [name] not found in Identity Store
```

Also, an error is logged containing more detailed information as shown here:

```
[2011-03-28T01:03:07.143-07:00] [WC_Spaces] [NOTIFICATION] [WCS-07855]
oracle.webcenter.doclib.internal.spaces.AbstractDoclibRoleMapper] [tid:
pool-1-daemon-thread-1] [userId: monty]
[ecid: a4789a41d7e6bc9f:36de4556:12efb72d049:-8000-00000000000002c0,0:5]
[APP: webcenter#11.1.1.4.0] Adding groups
[oracle.webcenter.security.common.WCGroup@18b96a3] to documents service roles
[Administration, Delete Documents, Create and Edit Documents, View Documents] for
 scope Scope[name=rbgs25mar01, guid=sbf125dd4_cd43_41cc_9d3d_467d06e84100]
[2011-03-28T01:03:09.122-07:00] [WC_Spaces] [ERROR] [WCS-44002]
[oracle.webcenter.security.rolemapping.RoleManager]
[tid: [ACTIVE].ExecuteThread: '3' for queue: 'weblogic.kernel.Default
(self-tuning)'] [userId: monty]
[ecid: a4789a41d7e6bc9f:36de4556:12efb72d049:-8000-00000000000002c0,0]
[APP: webcenter#11.1.1.4.0] The Role Mapping provider encountered an exception
while performing security role mapping for service oracle.webcenter.doclib.
[[oracle.webcenter.security.rolemapping.spi.RoleMappingSPIException: Cannot add
role null and permissions, 15, to the account for the folder, rbgs25mar01 for the
user/group Admin.        at
oracle.webcenter.doclib.internal.spaces.UCMSpacesUtils$2.newException(UCMSpacesUti
ls.java:2595)
```

Note: In previous releases, if a back-end server did not support enterprise groups, users belonging to enterprise groups were individually added to WebCenter Portal roles; this behavior has changed.

50 Customizing System Pages

This chapter describes the out-of-the-box system pages available in WebCenter Portal and how to customize them.

> **Note:** Any changes made to a system page at the application level by the system administrator are reflected in the equivalent system page in all portals. For example, if the system administrator adds an image to the application-level **Announcements** system page, that image will display in the portal-level **Announcements** system page in all portals. Changes made to a portal-level system page are reflected only in the associated portal. When displaying a system page in a portal, WebCenter Portal applies all customizations made at the application level, then customizations made at the portal level.

This chapter includes the following topics:

- Section 50.1, "About System Pages"
- Section 50.2, "Customizing System Pages for All Portals"
- Section 50.3, "Setting System Page Properties"
- Section 50.4, "Removing All Page Customizations from a System Page"

> **Permissions:** To perform the tasks in this chapter, you must have the WebCenter Portal `Administrator` role or a custom role that grants the following permission:
>
> - `Portal Server-Manage Configuration`
>
> For more information about permissions, see Section 49.3, "About Application Roles and Permissions."

50.1 About System Pages

System pages are provided out-of-the-box and are designed to fulfill a specific purpose. For example, users who are not logged in when they visit a portal may see the public **Welcome** page.

System pages include a variety of utility pages that you can customize to reflect your company brand, to provide useful hints, or to display other enhancements that you want. They support a rapid deployment of a portal and fulfill a range of needs, from providing an introductory page to pages that provide content that is generated

dynamically and tailored to the individual user (for example, the **Activity Stream** page).

System pages are preconfigured with page access settings that target their anticipated audience. For example, the **Welcome** page is configured to target the `anonymous-role`, the **Activity Stream** page is targeted to individual users, with dynamic content that is tailored to each user. In view of this preconfiguration, you cannot alter the security settings of a system page.

You can, however, customize system pages. Customization enables you to enhance the seeded content of a system page to apply your company brand, add hint text, provide additional functionality (such as task flows and portlets), and so on.

System pages also make task flow customization possible. The system page **Task Flow Editor** provides an environment for customizing all instances of a seeded task flow in a given scope in one operation. (Custom task flows created through the **Assets** or **Shared Assets** page cannot be edited using this page.) Authorized users can add seeded task flows to this page and then customize all instances of the task flow. For more information, see Chapter 54, "Customizing Task Flows Across Portals."

Table 50–1 lists and describes the system pages that are delivered out-of-the-box and provides information about the context in which they appear.

Table 50–1 Seeded System Pages

Page	Description	Context
Activities	For the Home portal, displays the Publisher task flow and the Activity Stream task flow from the People Connections service. For more information, see the "Tracking Portal Activities" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.	This is an application-level system page.
Activity Stream	For a portal, displays the Publisher task flow and the Activity Stream task flow from the People Connections service. For more information, see the "Tracking Portal Activities" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.	Appears by default in the Home portal of every authenticated (logged-in) user. Both application- and portal-level system pages are available.
Analytics	Provides information about application usage and performance metrics. For more information, see Chapter 56, "Analyzing Portal Usage." For Analytics task flows to work, the Analytics schema (`ACTIVITIES`) must be installed and configured, and a connection set up between WebCenter Portal and the Analytics Collector.	Available for showing in the Home portal. Useful only when configuration requirements are met. This is an application-level system page.
Announcements	Displays the Announcement Manager task flow. For more information, see the "Working with Announcements" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.	Appears by default in many out-of-the-box portal templates. Both application- and portal-level system pages are available.

Table 50–1 (Cont.) Seeded System Pages

Page	Description	Context
Discussions	Displays the Discussion Forum Manager task flow. For more information, see the "Viewing and Participating in Discussions" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.	Appears by default in many out-of-the-box portal templates. Both application- and portal-level system pages are available.
Documents	Displays the Document Explorer task flow. There are two **Documents** system pages: for the Home portal, which shows the current user's personal documents; and one for portals, which shows documents uploaded to that portal. For more information, see the "Creating and Managing Documents" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.	Appears by default in many out-of-the-box portal templates. Both application- and portal-level system pages are available.
Error Encountered	Displays an error page when an error occurs.	Appears when an application error occurs. This is an application-level system page.
Events	Displays the Events task flow. For more information, see the "Working with Calendars and Events" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.	Appears by default in many out-of-the-box portal templates. Both application- and portal-level system pages are available.
Lists	Displays the List Manager task flow. For more information, see the "Working with Lists" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.	Appears by default in many out-of-the-box portal templates. Both application- and portal-level system pages are available.
Login	Provides fields for logging in to your portal.	Appears instead of the WebCenter Portal Welcome Page when you disable public access to all application pages and when your current session expires. This is an application-level system page.
Members	Provides features for managing the members of a portal. For more information, see the "Managing Members and Assigning Roles in a Portal" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.	Appears in the default navigation as the **Members** page in some seeded portal templates. Both application- and portal-level system pages are available.
No Pages Accessible	Displays a message notifying the user that no pages are accessible.	Appears when users navigate to a portal in which they have no access permissions on the portal's pages. Both application- and portal-level system pages are available.

Table 50–1 (Cont.) Seeded System Pages

Page	Description	Context
Page Not Found	Displays a message notifying the user that the page cannot be found.	Appears when users navigate to a page that is no longer available in WebCenter Portal, or a page on which they do not have access permission. This is an application-level system page.
Page Viewer	Displays an external web site (such as google.com) in a portal, surrounded by the page template. For more information, see the "Adding Resources to a Navigation Model" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.	Used when you create a Navigation model that contains an External URL item (with target Same Page). When users click on such links in the navigation, the Page Viewer is used. Both application- and portal-level system pages are available.
Portal Not Found	Displays a message notifying the user that the portal cannot be found.	Appears when users navigate to a portal that is no longer available. This is an application-level system page.
Portals	Provides a view of all portals that the current user can access. Additionally provides features for creating, searching for, sorting, and filtering portals. For more information, see the "Viewing and Accessing Available Portals" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.	Appears when users select **Browse Portals** from the **Portals** menu. This is an application-level system page.
Portal Templates	Provides a view of all available portal templates. Includes controls for creating, editing, and filtering portal templates and viewing information about a selected portal template. For more information, see the "Working with Portal Templates" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.	Appears in the administration pages on the **Portal Templates** page. This is an application-level system page.
Profile	Displays the current user's Profile Gallery, which includes subpages for Activity Stream, Connections, Documents, an organization chart (Organization), and the user's profile details (About). For more information, see the "Managing Your Profile" and "Creating and Managing Documents" chapters in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.	Appears by default in the Home portal of every authenticated (logged-in) user. Both application- and portal-level system pages are available.

Table 50–1 (Cont.) Seeded System Pages

Page	Description	Context
Resource Viewer	Displays a resource in a portal, surrounded by the page template. For more information, see the "Adding Resources to a Navigation Model" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.	Used when you create a Navigation model that contains a resource item (with target Same Page). When users click on such links in the navigation, the Resource Viewer is used. Both application- and portal-level system pages are available.
Search	Displays the Search Results page. For more information, see the "About Searching in WebCenter Portal" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.	Renders dynamically to display the results of a search. Both application- and portal-level system pages are available.
Self-Registration	Provides a means of enabling users to create their own login accounts to your WebCenter Portal. For more information, see the "Registering Yourself with WebCenter Portal" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.	Appears when users click **Register** on the Login or WebCenter Portal Welcome Page. This is an application-level system page.
Self-Service Membership	Provides a means of subscribing to a portal that is configured to allow membership by subscription. For more information, see the "Joining a Portal" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.	Appears when users initiate a subscription to a portal. Both application- and portal-level system pages are available.
Tag Center	Displays the Tag Center task flow. For more information, see the "Using Tags and Bookmarks" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.	Appears when users navigate to the Tag Center by one of several methods. Both application- and portal-level system pages are available.
Task Flow Editor	Provides an environment for customizing all instances of a seeded task flow in a given scope in a single operation. (Custom task flows created through the Assets or Shared Assets page are not supported.) For more information, see Chapter 54, "Customizing Task Flows Across Portals."	Allows authorized users to add seeded task flows and then customize all instances of those task flows. Both application- and portal-level system pages are available.
Task Flow Viewer	Displays a task flow in a portal, surrounded by the page template. For more information, see the "Adding Resources to a Navigation Model" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.	Used when you create a Navigation model that contains a task flow item (with target Same Page). When users click on such links in the navigation, the Task Flow Viewer is used. Both application- and portal-level system pages are available.
Unauthorized	Displays a message notifying users that they are not authorized to access a portal or a page.	Appears when users navigate to a portal or a page on which they do not have access permission. Both application- and portal-level system pages are available.

Table 50–1 (Cont.) Seeded System Pages

Page	Description	Context
Unavailable	Displays a message notifying users that the portal is not available.	Appears when users navigate to a portal that is offline. Both application- and portal-level system pages are available.
User Profile	Displays the Profile Gallery of a user other than the current user, which, by default, displays the same subpages and task flows as the current user's Profile Gallery. It differs in that it displays information associated with the other users.	Renders dynamically when the current user accesses another user's profile. Both application- and portal-level system pages are available.
WebCenter Portal Welcome Page	Displays login fields, a registration link, boilerplate information with links to help topics about specific features, a link to public portals, and a language changer for the selection of an alternate session language.	This is the public welcome page. It is the first page users see when they access WebCenter Portal. If you decide to disable public access to all application pages, the public welcome page is not shown and users are directed to the Login page. This is an application-level system page.

50.2 Customizing System Pages for All Portals

You can customize out-of-the-box system pages to bring them in line with your organization's brand or look and feel. You can remove existing components, add new components, and change the page layout. You cannot, however, edit or delete system page input fields and buttons.

This section describes how to customize system pages for WebCenter Portal. It includes the following subsections:

- Section 50.2.1, "Customizing an Application-Level System Page"
- Section 50.2.2, "Creating a Page Variant of a System Page for Device Groups"
- Section 50.2.3, "Managing a Page Variant of a System Page for Device Groups"

To customize system pages for a portal, see the "Customizing System Pages in a Portal" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

50.2.1 Customizing an Application-Level System Page

You customize any system page that is editable. This procedure is for all application-level system pages. If a page variant of a system page for use by a device group has been created, you can also customize these page variants.

To customize an application-level system page or page variant:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **System Pages**.

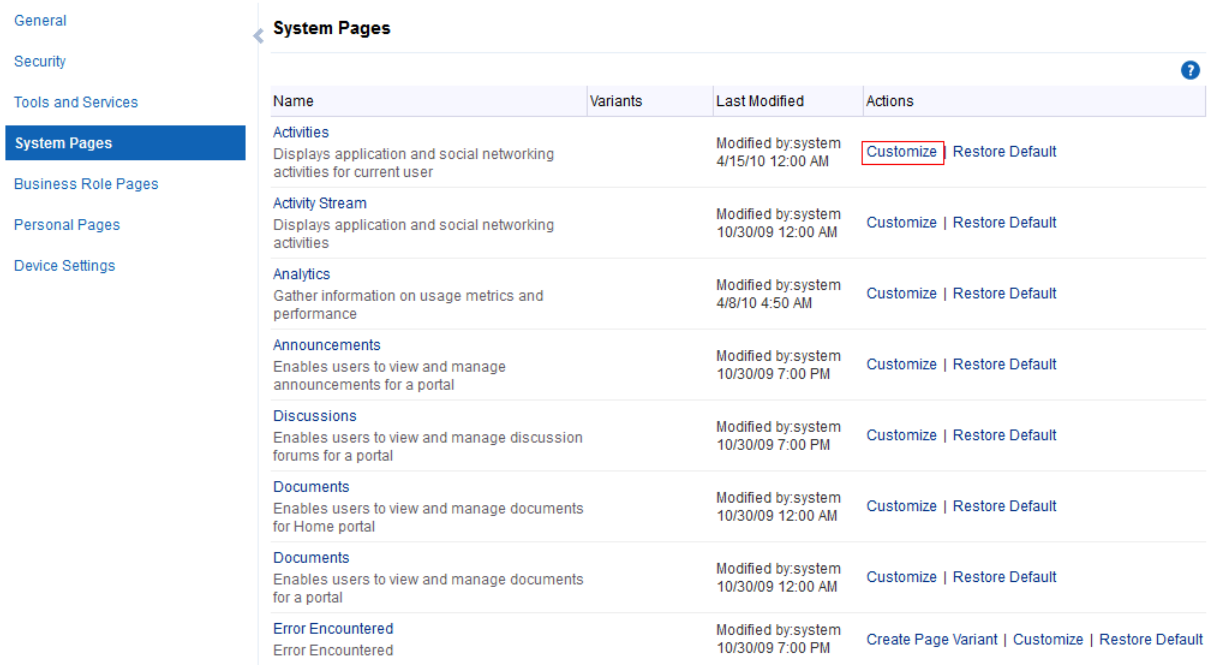
   You can also enter the following URL in your browser to navigate directly to the **System Pages** page:

   `http://host:port/webcenter/portal/builder/administration/systempages`

See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click the **Customize** link next to the system page to open it in Composer (Figure 50–1).

Figure 50–1 Customize Link Next to a System Page



3. To customize a variant of a system page for a device group (created by the system administrator at the application level), expand the system page variant icon, then click **Customize** for the device group you want to customize (Figure 50–2).

Figure 50–2 Customizing a System Page Variant for a Device Group

4. Customize and save the page.

See Also: For information about editing pages, see the "Editing a Page" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

50.2.2 Creating a Page Variant of a System Page for Device Groups

Page variants are alternative views of an existing page for specific device groups to target specific device size and characteristics. The base page and the page variant have the same URl and security settings; however, any content changes to the base page is not reflected in the variant pages and vice versa.

> **Note:** For more information about managing device settings in WebCenter Portal, see Chapter 53, "Administering Device Settings."

Out-of-the-box, you can create page variants for the following system pages only: **Error Encountered**, **Login**, **No Pages Accessible**, **Page Not Found**, **Portal Not Found**, **Self-Registration**, **Unauthorized**, **Unavailable**, and **WebCenter Portal Welcome Page**.

Only system administrators can create page variants for the application-level system pages. A portal moderator can customize a page variant for a system page at the portal-level only after the system administrator has created a page variant for that system page. To customize a system page or page variant for a portal, see the "Customizing System Pages for a Portal" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

If a page variant is not created for a supported device group, then the base page displays only devices that belongs to that device group.

You can create a page variant for each device group that is available. However, you can create only one page variant for a device group per page. In other words, you cannot create two page variant for the iOS Phones device group for the same page, but you can create a page variant for the iOS Phones device group and another page variant for the Android Phones device group for the same page. You can create a page variant for the iOS Phones device group for a different page.

To create a page variant of a system page for device groups:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **System Pages**.

   You can also enter the following URL in your browser to navigate directly to the **System Pages** page:

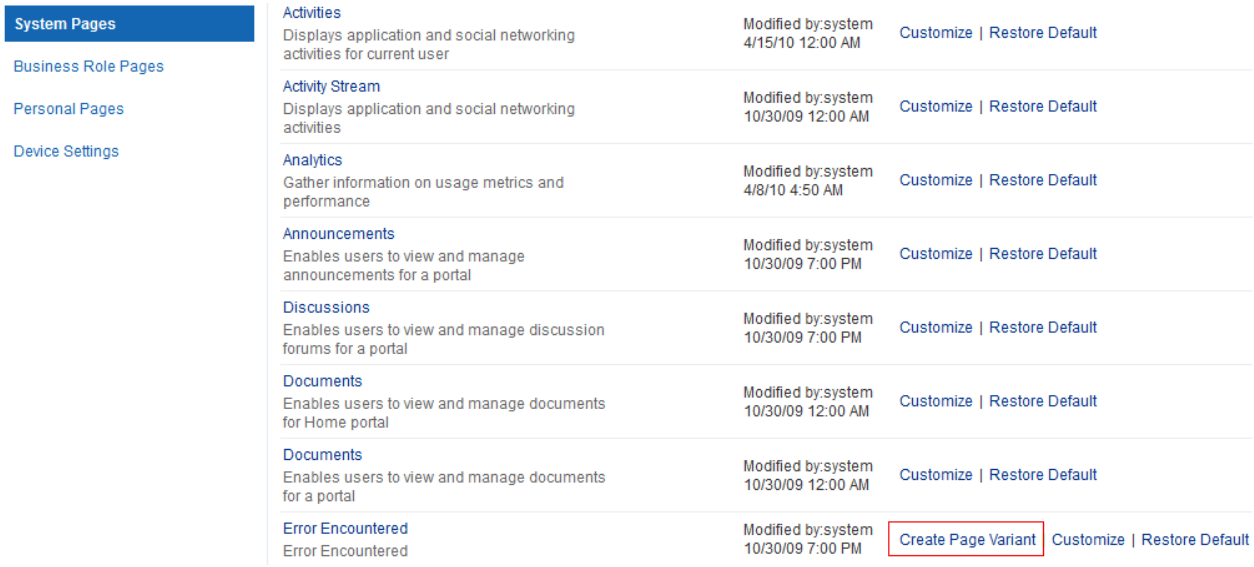
   `http://host:port/webcenter/portal/builder/administration/systempages`

   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click the **Create Page Variant** link next to the system page (Figure 50–3) for which you want to create a page variant.

   > **Note:** Out-of-the-box, page variants for device groups are provided for the following system pages only: **Error Encountered**, **Login**, **No Pages Accessible**, **Page Not Found**, **Portal Not Found**, **Self-Registration**, **Unauthorized**, **Unavailable**, and **WebCenter Portal Welcome Page**.

Figure 50–3 Create Page Variant Link Next to a System Page



3. In the Create Page Variant dialog that appears, select the device group for which you want to create a page variant from the **Device Group** drop-down list (Figure 50–4)

   The base page is seeded in the system. The base page is always rendered for devices belonging to the default device group (for more information about the default device group, see Chapter 53, "Administering Device Settings"). If a page variant exists for a device group that is also set as default, then the base page will take precedence over the page variant. By default the device group is set to **Desktop Browser**s if you open a page from your desktop browser, so you still see the base page, whether or not the **Desktop Browsers** variant is created. From other devices, you will see the page variant you select.

   For example, if you change **iOS Phones** to the default page, the base page is set for that device type. On an iphone, the base page is displayed and not the **iOS Phones** page variant. However, on the desktop, the **Desktop Browser**s variant is displayed, not the base page. If you do not change the default device group, the **Desktop Browser**s variant that is created will not display on desktop browsers. The base page will still display on the desktop.

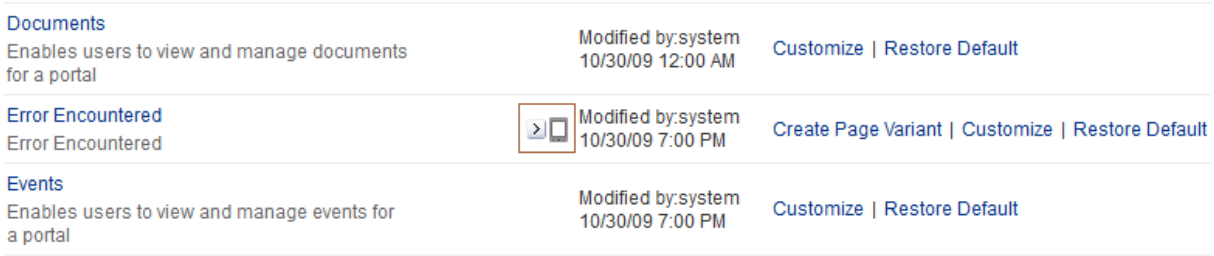
   Note: Use caution if you change the default device group—it will change the default behavior when globally displaying base pages or their page variants.

Figure 50–4 Create Page Variant Dialog

4. Click **Create**.

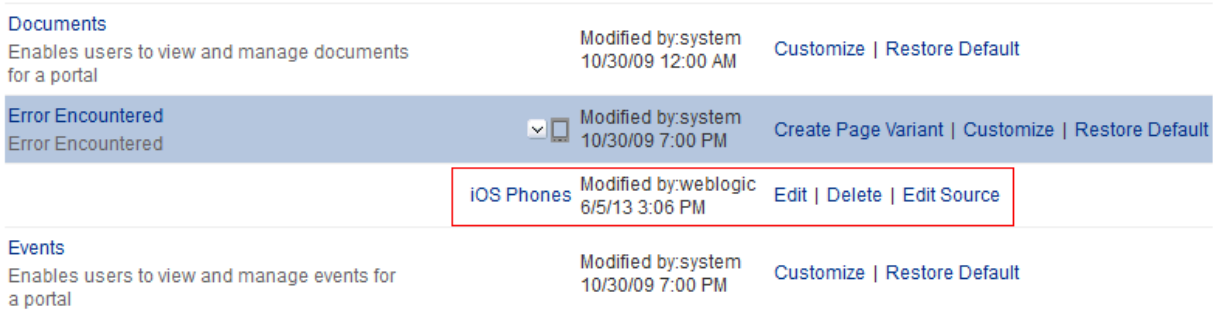
   A mobile icon appears next to the page, indicating that a page variant for the page is available (Figure 50–5).

Figure 50–5 Icon Showing That a Page Variant is Available



5. Click the **Expand** icon to view the device group page variant (Figure 50–6).

Figure 50–6 Page Variant for a Device Group



   You can create another page variant for another device group for the same page. However, you cannot create another page variant for the same device group that already has a page variant.

6. You can do any of the following after creating a page variant:

   - Click **Edit** next to the device group to edit the page in Composer.

     For information about editing pages, see the "Editing a Page" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

   - Click **Delete** next to the device group to delete the page variant. Confirm the deletion by clicking **Delete** again.

- Click **Edit Source** next to the device group to edit the source code.

  For more information, see the "Viewing and Modifying Page Source Code" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

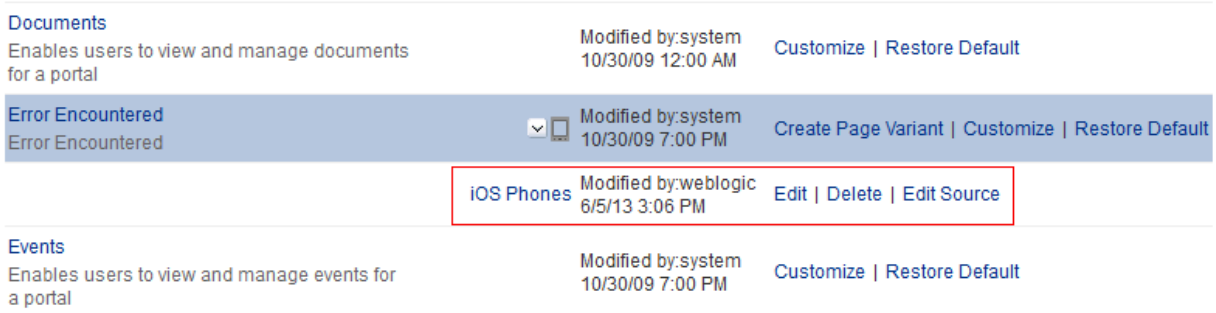
50.2.3 Managing a Page Variant of a System Page for Device Groups

For information about page variants for device groups and creating a page variant for a system page, see Section 50.2.2, "Creating a Page Variant of a System Page for Device Groups."

To manage a page variant of a system page:

1. Click the **Expand** icon to view the device group page variant (Figure 50–7).

Figure 50–7 Page Variant for a Device Group



2. To edit the page variant in Composer, click **Edit**.

   For information about editing pages, see the "Editing a Page" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

3. To delete the page variant, click **Delete**. Confirm the deletion by clicking **Delete** again.

4. To edit the source code, click **Edit Source**.

   For information about editing page source code, see the "Viewing and Modifying Page Source Code" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

50.3 Setting System Page Properties

The page properties for system pages provide a means of specifying a page background color and image, applying additional CSS encoding, and setting parameters.

To edit the properties of a system page:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **System Pages**.

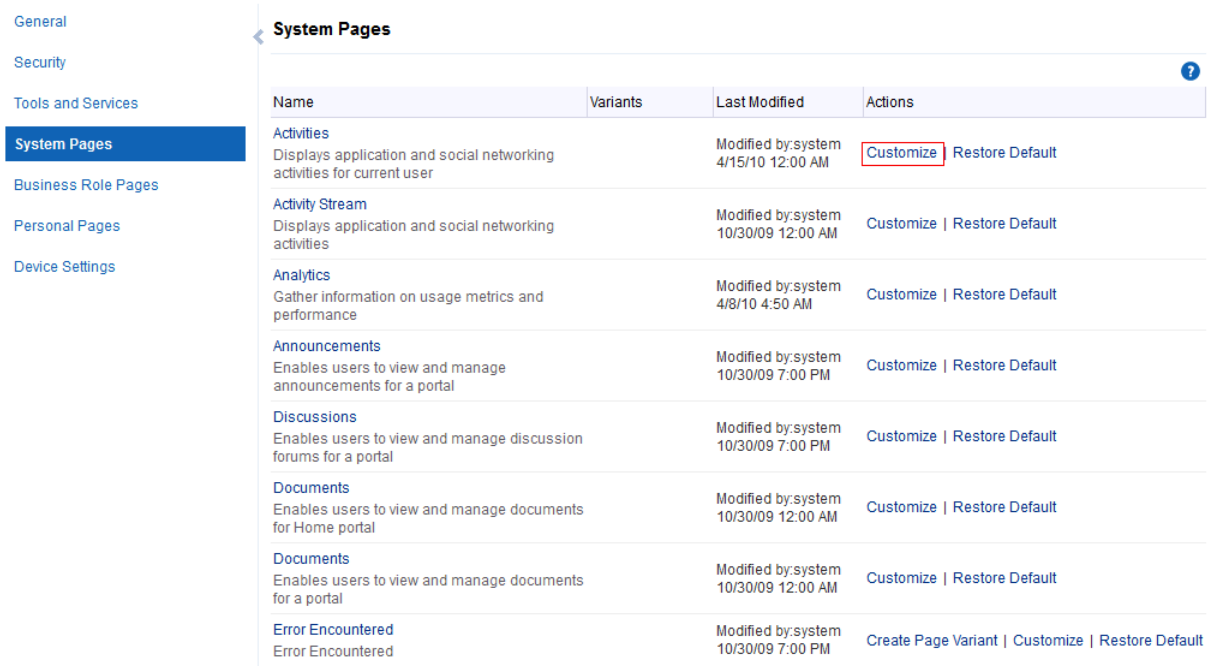
   You can also enter the following URL in your browser to navigate directly to the **System Pages** page:

   `http://host:port/webcenter/portal/builder/administration/systempages`

   See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click the **Customize** link next to a system page to open it in Composer (Figure 50–8).

Figure 50–8 Customize Link Next to a System Page



3. Click the **Page Properties** icon (Figure 50–9) to open the Page Properties dialog (Figure 50–10).

See Also: For information about editing pages, see the "Editing a Page" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Figure 50–9 Page Properties Icon

4. To change properties on the **Display Options** tab, see the "Providing Page Background Color, Image, and CSS Encoding" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

5. On the **Parameters** tab (Figure 50–10), modify existing parameters as required (see Table 50–2).

Figure 50–10 Page Properties Dialog: Parameters

Note: All parameter values provide access to an Expression Language (EL) editor, which you can use to select or specify a variable value instead of a constant value. Click the **Edit** icon next to a value field, then select **Expression Builder** to open the editor. If you need EL assistance, an application developer can provide an EL expression; see the "Expression Language Expressions" appendix in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

Table 50–2 System Page Parameters

Parameter	Description
`pg_theme`	Specifies whether to display the system page using the out-of-the-box look and feel of prior releases, or the new look and feel of the current release. Valid values are: ▪ `spaces_blue`. Use look and feel of prior releases (11.1.1.7.0 and earlier). ▪ `11ps7_grey` or leave this field blank. Use new look and feel of current release.

Table 50–2 (Cont.) System Page Parameters

Parameter	Description
`pg_pageTemplateID`	Specifies whether to display the system page using the default page template of prior releases, or the default blank template of the current release. Valid values are: ▪ `spaces_blue`. Use default page template of prior releases (11.1.1.7.0 and earlier). ▪ Leave this field blank. Use default blank template of current release. ▪ *`page_template_GUID`*. Use page template specified by GUID value.

6. To add a new parameter:
   - Click **Add a page parameter**.
   - In the Add a Page Parameter dialog, enter a new parameter **Name**, then click **Add Parameter** to add the parameter to the **Parameters** tab, with a value entry field.
   - Optionally, enter a value for the new parameter.

50.4 Removing All Page Customizations from a System Page

You can return a system page to its default, out-of-the-box state, removing all page customizations.

> **Note:** This process does not remove task flow customizations. To remove task flow customizations, you must revise the given task flow on a system page. For more information, see Chapter 54, "Customizing Task Flows Across Portals."

To remove all customizations from a system page:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **System Pages**.

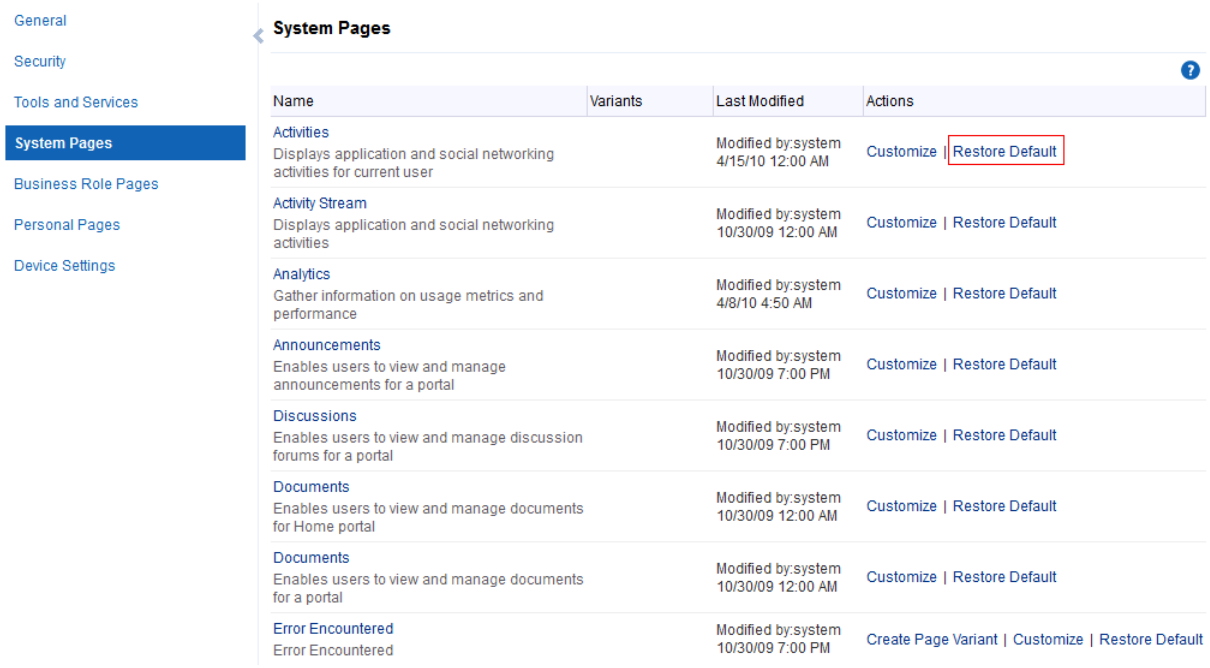
   You can also enter the following URL in your browser to navigate directly to the **System Pages** page:

   `http://host:port/webcenter/portal/builder/administration/systempages`

   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click the **Restore Default** link next to the system page (Figure 50–11).

Figure 50–11 Restore Default Link Next to a System Page



3. In the resulting confirmation dialog, click **Restore**.

   All customizations are permanently removed from the selected system page. When you restore a system page to its default state, page variants are not affected if the system page has variants.

51

Managing Business Role Pages

This chapter describes how to create and target business role pages and how to perform other related business role page management tasks.

Business role pages provide a means of exposing highly relevant content to a specific audience. Business role pages are pages targeted to a particular type of group, or user (or user role), such as your sales force, your accounting team, your administrative staff, and so on. Business role pages are exposed to targeted users in their views of the Home portal.

This chapter includes the following topics:

Permissions: To perform the tasks in this chapter, you must have the WebCenter Portal `Administrator` role or a custom role that grants the following permission:

- `Portal Server-Manage Configuration`

Additionally, you need permissions to create and manage portals (`Portals-Create` and/or `Portals-Manage All`). For more information about permissions, see Section 49.3, "About Application Roles and Permissions."

51.1 About Business Role Pages

A business role page may be available in the Home portal views of all users who share the targeted business role when the WebCenter Portal system administrator sets up a navigation model that publishes business role pages. For example, a business role page that targets all users assigned the `HR_ORG` role appears in the Home portal views of all users assigned the role `HR_ORG`.

> **Tip:** Whether or not a business role page is shown in the Home portal navigation, it is always available to targeted users on the **Personalize Pages** page.

If an individual user who is not assigned the `HR_ORG` role wants to see the page, the system administrator can grant access to this user. Seeded business role pages (see Table 51–1) have preconfigured access settings that cannot be altered. For information about how to alter access settings on seeded business role pages, see Section 51.4.3, "Setting Access on a Seeded Business Role Page."

The system administrator is the only type of user who can create a business role page. Only when a system administrator grants permission to do so, can other users edit, copy, and delete business role pages and change page permissions (for more information, see Section 51.4, "Specifying the Target Audience for a Business Role Page").

Table 51–1 lists and describes the seeded business role pages included in a default WebCenter Portal installation and provides information about the context in which they appear.

These pages, with the exception of WebCenter Portal Impersonation, also appear on the **System Pages** subtab (for more information, see Chapter 50, "Customizing System Pages").

Table 51–1 Seeded Business Role Pages

Page	Description	Context
Activities	Displays the Activity Stream from People Connections and a Publisher task flow, which can be used to post content to the stream. For more information, see the "Tracking Portal Activities" section and the "Working with Feedback and the Message Board" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.	Appears by default in the Home portal views of each authenticated (logged-in) user.
Analytics	Displays performance metrics related to WebCenter Portal, portals, portlets, and services. For more information, see Section 27.1, "Understanding Oracle WebCenter Portal Performance Metrics."	Is hidden by default, but can be accessed on the **Personalize Pages** page in the system administrator's view of the Home portal.
Documents	Displays the Document Explorer task flow. For more information, see the "About the Documents Service Task Flows" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.	Appears by default in the Home portal views of each authenticated user.
Profile	Displays the current user's Profile, which includes subpages for Activities, Connections, Documents, organization chart (Organization), and the user's profile details (About). For more information, see the "Managing Your Profile" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal* and the "About the Documents Service Task Flows" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.	Appears by default in the Home portal views of each authenticated user.

Table 51–1 (Cont.) Seeded Business Role Pages

Page	Description	Context
Portals	Displays portals relevant to the current user, such as the portals to which the user belongs or has access, and the portals that the user can search for. Each listed portal has an associated menu with options for performing actions on the portal. This page also provides controls for creating portals and searching for additional portals.	Appears by default in the Home portal views of each authenticated user.
Portal Templates	Displays a list of default and custom portal templates and provides a means of creating custom portal templates and filtering the template list.	Is hidden by default, but can be accessed on the **Personalize Pages** page in the Home portal views of each authenticated user.
Tag Center	Displays the Tag Center that is rendered when users click a tag in a Tags task flow or in search results. For more information, see the "Working with Tags" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.	Invoked when users click tags.
WebCenter Portal Impersonation	Displays a page, from where a WebCenter Portal system administrator can assign impersonation rights to a group of users ("impersonators"), such as support representatives or other portal administrators, so that they can perform operations as other users ("impersonatees"). For more information, see Chapter 38, "Managing Impersonation." For instructions on how to initiate an impersonation session (by the impersonator) and how to allow an Impersonation session (by the impersonatee), see the "Using WebCenter Portal Impersonation" chapter in the *Oracle Fusion Middleware Using Oracle WebCenter Portal*. For information about impersonation ELs and APIs, see the "Using WebCenter Portal Impersonation ELs and APIs" section in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.	WebCenter Portal impersonation relies on OAM 11*g* 11.1.2.0. Before you can enable impersonation for a WebCenter Portal instance, you must first install and configure OAM 11g (Oracle's single sign-on solution), and then turn on impersonation in OAM. For information about installing and configuring OAM 11*g*, see Section 33.2, "Configuring Oracle Access Manager (OAM)."

51.2 Setting Page Creation Defaults for Business Role Pages

As the WebCenter Portal system administrator, you can set page creation defaults to reduce the number of steps required to create business role pages. That is, you can specify the page style that is selected by default when you open the Create Page dialog. You can also select to bypass the Create Page dialog, which enforces the default page style.

> **See Also:** The page creation defaults that the system administrator sets for business role pages also affect personal pages. Authorized users can override page creation defaults for their own personal pages created in the Home portal (for more information, see the "Setting Page Creation Defaults for Personal Pages" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*). Defaults for pages created in a portal are controlled by the portal moderator (for more information, see the "Creating and Editing a Portal Page" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

To set page creation defaults for business role pages:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Business Role Pages**.

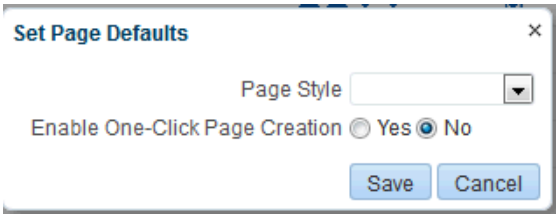
   You can also enter the following URL in your browser to navigate directly to the **Business Role Pages** page:

```
http://host:port/webcenter/portal/builder/administration/businessrolepages
```

> **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click **Set Page Defaults** to open the Set Page Defaults dialog (Figure 51–1).

Figure 51–1 Set Page Defaults Dialog



3. Select a page layout from the **Page Style** drop-down list.

> **See Also:** For information about the seeded page styles, see the "Renaming a Page" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*. The list may include additional custom page styles or restrict page styles to a shorter list.

4. Select an option for **Enable One-Click Page Creation**:
   - **Yes**: Bypass the Create Page dialog, and create all of your personal pages using the specified **Page Style**.

     > **Tip:** When you create pages by bypassing the Create Page dialog, the new page has a generic name. For information about renaming pages, see the "Renaming a Page" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

   - **No**: Display the Create Page dialog, with the specified **Page Style** selected as the default in the Create Page dialog for all of your personal pages. You can select a different style for your new personal pages.
5. Click **Save** to save your changes and exit the dialog.

51.3 Creating a Business Role Page

> **Note:** You can also select the **Copy Page** action for a Personal Page or a Business Role page and select to copy it as a Business Role page. For more information, see Section 52.8, "Copying a Personal Page" and Section 51.10, "Copying a Business Role Page."

To create a new business role page:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Business Role Pages**.

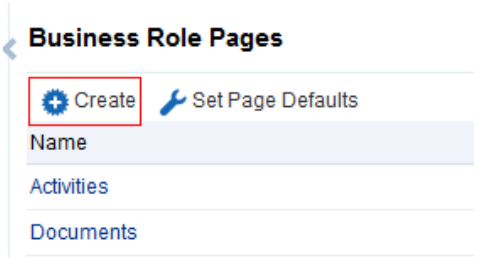
   You can also enter the following URL in your browser to navigate directly to the **Business Role Pages** page:

   ```
   http://host:port/webcenter/portal/builder/administration/businessrolepages
   ```

See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click **Create** (Figure 51–2).

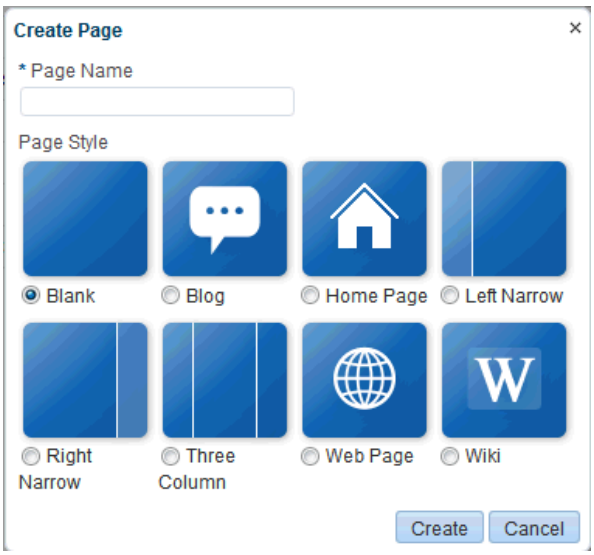
Figure 51–2 Create Option for a Business Role Page



If you enabled one-click page creation, the new page appears in the list. If you did not enable one-click page creation, continue with the next steps.

3. In the **Create Page** dialog, enter a unique name for the page in the **Page Name** field, and then select a **Page Style** (Figure 51–3).

Figure 51–3 Create Page Dialog



See Also: For an overview of seeded page styles, see the "Out-of-the-box Page Styles" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

4. Click **Create**.

The new page appears in the list of Business Role pages.

See Also: The system administrator can set an attribute on a custom page style that determines whether a newly created page that is based on that style opens in page edit mode or page view mode. For more information, see the "Setting Properties on a Portal Asset" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Later, you can add content to the page. The next section (Section 51.4, "Specifying the Target Audience for a Business Role Page") steps you through setting access permissions for the business role page.

5. Next steps:
   - Define the page audience, as described in Section 51.4, "Specifying the Target Audience for a Business Role Page."
   - Choose the page display order, as described in Section 51.7, "Setting a Default Display Order for Business Role Pages."
   - Add content to the page, as described in the sections "Adding and Configuring Page Layout Components" and "Adding Portal Components to a Page Template" in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

51.4 Specifying the Target Audience for a Business Role Page

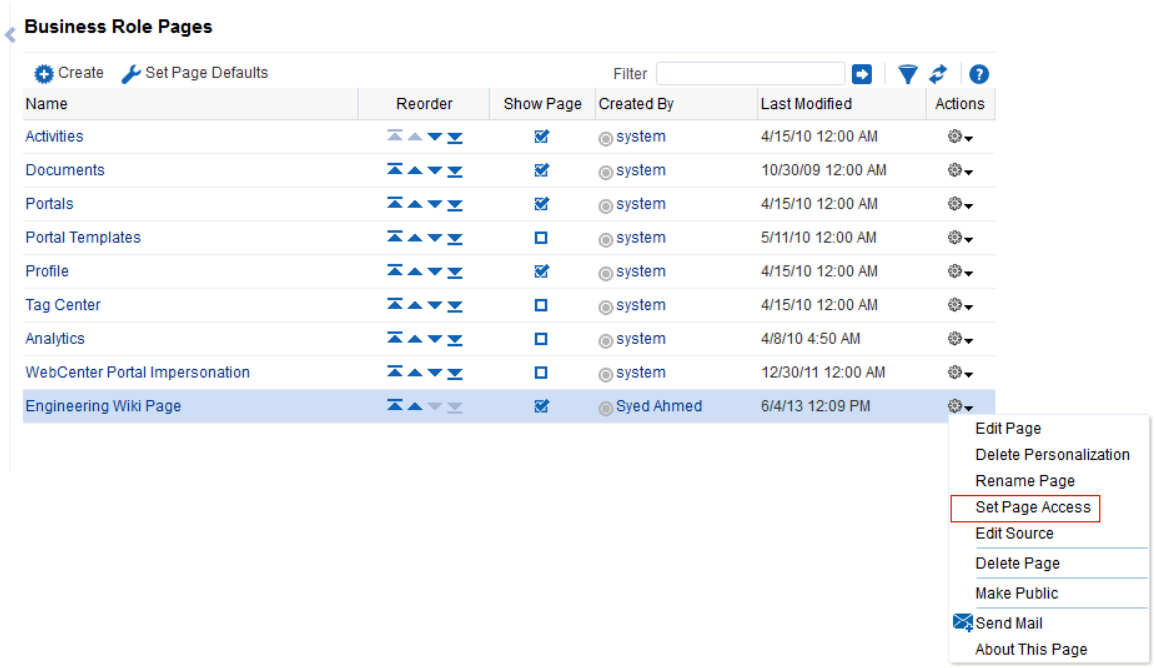
The target audience for business role pages may change from time to time. For example, you may want the whole Sales team to see a page originally designed for a Product Development team. You may want to provide public access to the Marketing department's page. You may want to provide additional access privileges, such as the *Edit Page* privilege, to a selected department member.

Note: As the system administrator, you can set access on the business role pages that you create (for more information, see Section 51.4.1, "Setting Access on a Custom Business Role Page").

You cannot alter the default access settings of seeded business role pages through the WebCenter Portal user interface (see Table 51–1, " Seeded Business Role Pages"). For information about how to set access on seeded business role pages, see Section 51.4.3, "Setting Access on a Seeded Business Role Page."

You can find controls for setting page access for a **Personalize Pages** page in the Home portal, and for a **Business Role Pages** page in WebCenter Portal Administration (Figure 51–4).

Figure 51–4 Set Page Access Option on a Custom Business Role Page



This section describes how to set specific access on a business role page as well as how to make such a page public. It includes the following subsections:

- Section 51.4.1, "Setting Access on a Custom Business Role Page"
- Section 51.4.2, "Providing Public Access to a Custom Business Role Page"
- Section 51.4.3, "Setting Access on a Seeded Business Role Page"

51.4.1 Setting Access on a Custom Business Role Page

As the system administrator, you can use the WebCenter Portal Administration user interface to set access on the business role pages that you create. However, you cannot use the WebCenter Portal Administration user interface to set access on seeded business role pages (for information about setting access on seeded business role pages, see Section 51.4.3, "Setting Access on a Seeded Business Role Page").

To specify the target audience for a custom business role page that you created:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Business Role Pages**.

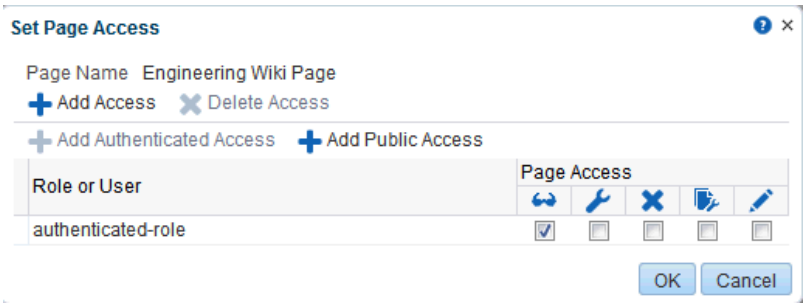
   You can also enter the following URL in your browser to navigate directly to the **Business Role Pages** page:

   `http://host:port/webcenter/portal/builder/administration/businessrolepages`

   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Select and click the **Actions** icon for the custom business role page for which you are setting access, and select **Set Page Access** to open the Set Page Access dialog (Figure 51–5).

Figure 51–5 Set Page Access Dialog



The `authenticated-role` role is added under **Role or User**, that is users who are logged in to the WebCenter Portal.

3. To grant access permissions to all public users, that is, users who have not logged in (as well as those who have), click **Add Public Access**.

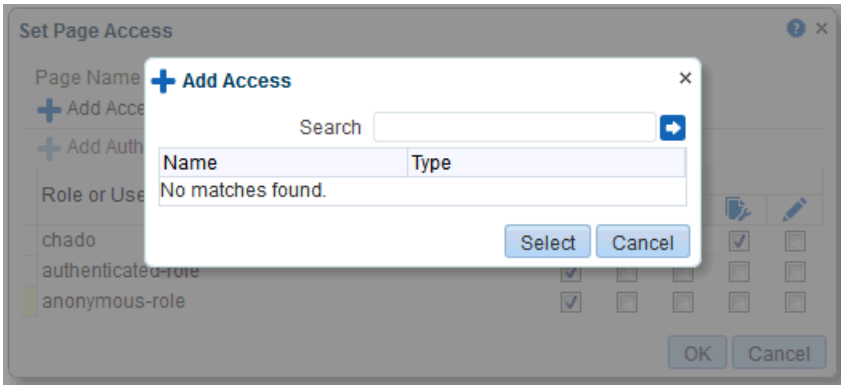
   The `anonymous-role` role is added under **Role or User**.

   > **See Also:** This method of enabling public access to a business role page provides a means of granting more than view access to public users. If you want to give such users view access only, you can set this quickly by following the steps in Section 51.4.2, "Providing Public Access to a Custom Business Role Page."

4. To grant access permissions to selected users, groups, and application roles, click **Add Access**.

   The Add Access dialog opens (Figure 51–6).

Figure 51–6 Add Access Dialog



5. Identify the users, groups, and application roles for whom to expose this business role page in the Home portal.

   Choose from all available users, groups, and application roles. Use the Search feature to search your identity store:

   a. In the **Search** field, enter a search term for a user, group, or application role. For tips on searching the identity store, see the "Searching for a User or Group in the Identity Store" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

      > **Tip:** This search is not case sensitive.

b. Click the **Search** icon.

   Users, groups, and application roles matching your search criteria appear in the **Add User** dialog.

c. Select one or multiple names from the list.

   Press `Ctrl+Click` to select multiple users.

d. Click **Select**.

   The results of your selection appear in the Set Page Access dialog. By default, selected users have the *View Page* permission.

6. For each user name, group, or application role, select one or more check boxes to grant page privileges:

   - **View Page**—Users can view the page, but cannot perform any actions on the page.
   - **Edit Page**—Users can edit the page. This includes adding, rearranging, and deleting content, and changing the page scheme.
   - **Delete Page**—Users can delete the page.
   - **Perform All Page Actions**—Users have full access rights to the page. These users can edit the page, revise the page layout, set additional access privileges for other users, and all other page privileges.
   - **Personalize Page**—Users can change their personal view of the page. Such changes do not affect another user's view of the page.
   - To revoke access to the page, select the user or role and click **Delete Access**.

     Tip: To revoke a privilege, deselect the check box.

7. Click **OK** to save your changes.

   The page is displayed to its target audience, who can see it in their views of the Home portal the next time they log in to WebCenter Portal.

51.4.2 Providing Public Access to a Custom Business Role Page

You can specify that any user, whether logged in or not, can view a particular custom business role page. Such a page can be exposed in a public Home portal, or you can publish the URL to the public business role page to provide all users easy access.

> **See Also:** The process described in this section enables all public users to view a selected custom business role page. To provide public users with additional permissions on the page, follow the steps described in Section 51.4.3, "Setting Access on a Seeded Business Role Page."

To make a custom business role page public:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Business Role Pages**.

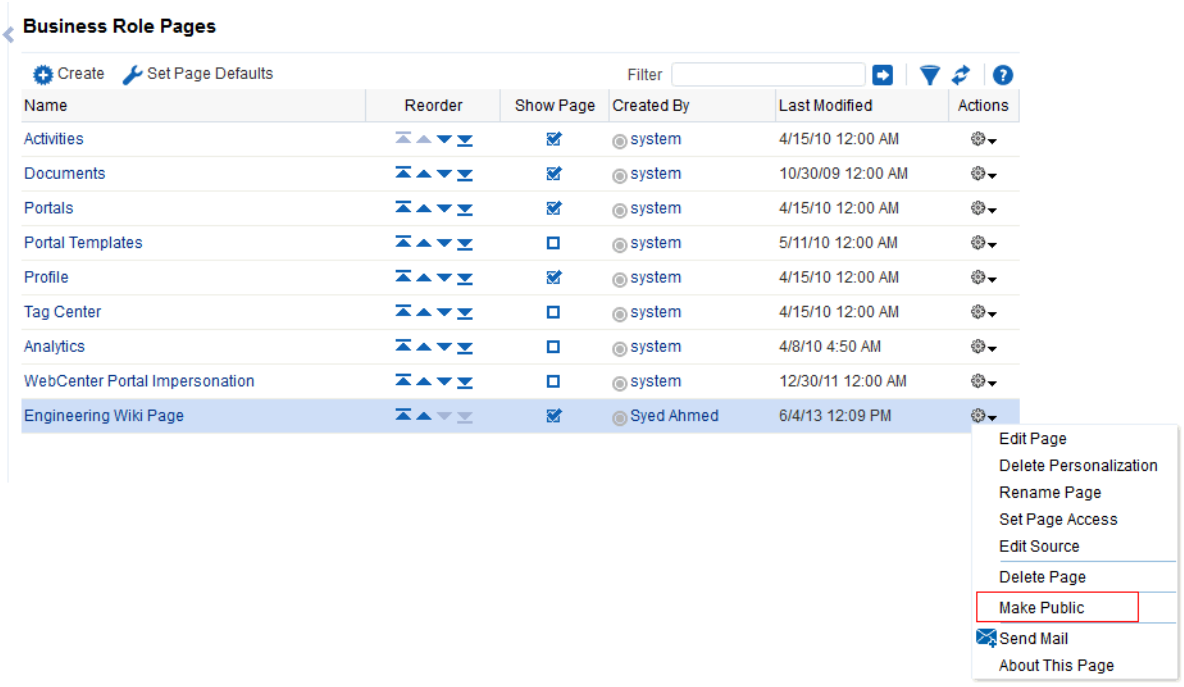
   You can also enter the following URL in your browser to navigate directly to the **Business Role Pages** page:

   `http://host:port/webcenter/portal/builder/administration/businessrolepages`

See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click the **Actions** icon for the business role page for which you are setting access, and select **Make Public** (Figure 51–7).

Figure 51–7 Make Public Option on a Custom Business Role Page



51.4.3 Setting Access on a Seeded Business Role Page

Out-of-the-box business role pages, such as Activities and Portals, are available to all users by default (see Table 51–1, " Seeded Business Role Pages"). You cannot modify the security of seeded business role pages through WebCenter Portal. If you want to change the default security settings, for example, you want to hide a seeded business role page from all users, you must modify the default business role page settings in `pages.xml` file, and upload the changes to the MDS repository used by WebCenter Portal using the WLST commands `exportMetadata` /`importMetadata`.

To modify the default security settings for a seeded business role page:

1. Run the WLST command `exportMetadata` to export `pages.xml` for the following user roles: `anonymous-role` and `authenticated-role`.

   For example:

```
exportMetadata(application='webcenter',server='WC_Spaces',toLocation='/scratch/
mdsdump',
docs='/oracle/webcenter/page/scopedMD/s8bba98ff_4cbb_40b8_beee_296c916a23ed/rol
e/anonymous-role/pages.xml')

exportMetadata(application='webcenter',server='WC_Spaces',toLocation='/scratch/
mdsdump',
docs='/oracle/webcenter/page/scopedMD/s8bba98ff_4cbb_40b8_beee_296c916a23ed/rol
e/authenticated-role/pages.xml')
```

Where `toLocation` specifies a target directory on your system for the file you want to export. For detailed syntax, see the section "exportMetadata" in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

2. Modify the security in both `pages.xml` files as required, that is, mark each business role page as hidden or shown:

```
<!-- Business Role Pages -->
     <pageDef id="Page_2eb852ac_10f5902cb2f__7ff7"
     contentMRef="/oracle/webcenter/page/scopedMD/
     s8bba98ff_4cbb_40b8_beee_296c916a23ed/businessRolePages/
     ActivityStreamMainView.jspx" shared="true" hidden="true"/>...

     <pageDef id="Page_2eb852ac_10f5902cb2f__7ff7"
     contentMRef="/oracle/webcenter/page/scopedMD/
     s8bba98ff_4cbb_40b8_beee_296c916a23ed/businessRolePages/
     ASpaceTemplatesMainView.jspx" shared="true" hidden="false"/>...

     <pageDef id="Page_2eb852ac_10f5902cb2f__7ff7"
     contentMRef="/oracle/webcenter/page/scopedMD/
     s8bba98ff_4cbb_40b8_beee_296c916a23ed/businessRolePages/
     MyProfileMainView.jspx" shared="true" hidden="true"/>...
```

   - Set `hidden="true"` for the pages that should be hidden.
   - Set `hidden="false"` for the pages that should be shown.

3. Upload your changes to the `pages.xml` files to MDS using the WLST command `importMetadata`.

   For example:

```
importMetadata(application='webcenter',server='WC_Spaces',fromLocation='/scratc
h/mdsdump',
docs='/oracle/webcenter/page/scopedMD/s8bba98ff_4cbb_40b8_beee_296c916a23ed/rol
e/anonymous-role/pages.xml')
importMetadata(application='webcenter',server='WC_Spaces',fromLocation='/scratc
h/mdsdump',
docs='/oracle/webcenter/page/scopedMD/s8bba98ff_4cbb_40b8_beee_296c916a23ed/rol
e/authenticated-role/pages.xml')
```

   Where `fromLocation` specifies the directory that contains the file you want to import. For detailed syntax, see the section "importMetadata" in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

   For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

Note: By default, any authenticated or annonymous user role will not be able to view the Activity Stream page (used as an example here). However, if the user logs into WebCenter Portal, from the **Personalize Pages** page the user can override this setting and make the page visible using the **Show Page** option. This user customization will be stored in MDS too, as `/oracle/webcenter/page/scopedMD/s8bba98ff_4cbb_40b8_beee_296c916a23ed/user/<GUID of user>/pages.xml`

The *<GUID of user>* can be queried from the table `WC_AS_ACTOR_DETAIL.ACTOR_ID`.

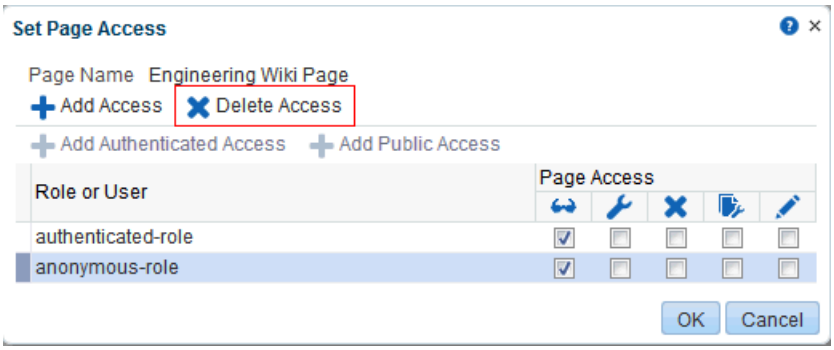
If you delete this `pages.xml` file within MDS, then it would revert to the set functionality from `/oracle/webcenter/page/scopedMD/s8bba98ff_4cbb_40b8_beee_296c916a23ed/role/authenticated-role/pages.xml`.

51.5 Revoking Access to a Custom Business Role Page

To revoke access privileges to a custom business role page:

1. Follow the steps in Section 51.4.1, "Setting Access on a Custom Business Role Page" to open the Set Page Access dialog.
2. From Role or User, select the row that has user, group, or application role from whom you want to revoke access, and click **Delete Access** (Figure 51–8).

Figure 51–8 Delete Access Option in Set Page Access Dialog



3. Click **Delete** in the confirmation dialog.

51.6 Providing Navigation to Business Role Pages

You have the following options for providing your users with navigation to business role pages:

- You can make the page visible to all authorized users in one step, using the **Show Page** check box.
- You can create a navigation link to a business role page.

This section describes how to use the **Show Page** option. It also provides a pointer to information about creating your own navigation.

This section includes the following subsections:

- Section 51.6.1, "Showing and Hiding Business Role Pages"
- Section 51.6.2, "Creating Navigation to a Business Role Page"

51.6.1 Showing and Hiding Business Role Pages

A control is available in the WebCenter Portal Administration pages for instantly showing or hiding a business role page to all authorized users (authorized users are users who have been granted access to the business role page). Use the **Show Page** check box to expose the page in the navigation of all authorized users.

To show or hide a business role page in Home portal navigation:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Business Role Pages**.

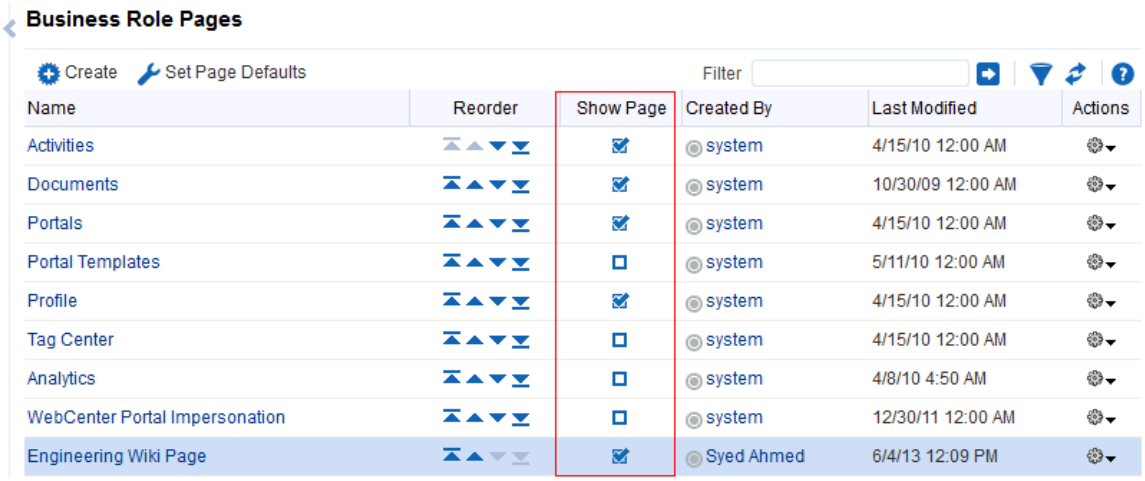
   You can also enter the following URL in your browser to navigate directly to the **Business Role Pages** page:

   `http://host:port/webcenter/portal/builder/administration/businessrolepages`

   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. For the page you want to show or hide (Figure 51–9):
   - Select the check box in the **Show Page** column to show the page in the Home portal views of authorized users.
   - Deselect the check box in the **Show Page** column to hide the page from view.

Figure 51–9 Show Page Option for Business Role Pages



51.6.2 Creating Navigation to a Business Role Page

After you make a business role page available to users through page permissions, you will likely also want to make it easy to access by including a link to it in your WebCenter Portal navigation. This process is described in detail in the "Adding Resources to a Navigation Model" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

51.7 Setting a Default Display Order for Business Role Pages

If you present business role pages in a logical order, the page content is more accessible and easier for users to navigate. As the WebCenter Portal system administrator, you can determine the initial order in which business role pages are presented to their intended audience. You can do this by dragging and dropping pages into the desired order or by clicking the **Reorder** icons.

Individual users can change the initial display order you specify on their **Personalize Pages** page in the Home portal. Additionally, they can hide the business role pages they do not use.

> **Note:** There are two locations from which to define the order and the visibility of pages: from WebCenter Portal administration (described here) and from the **Personalize Pages** page (described in the "Rearranging Page Order in the Home Portal" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*. The difference between the two is that the administration change is an *application customization* and the **Personalize Pages** change is a *user customization*. Keep in mind that user customizations override application customizations in a given user's view.

To change the display order of all business role pages:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Business Role Pages**.

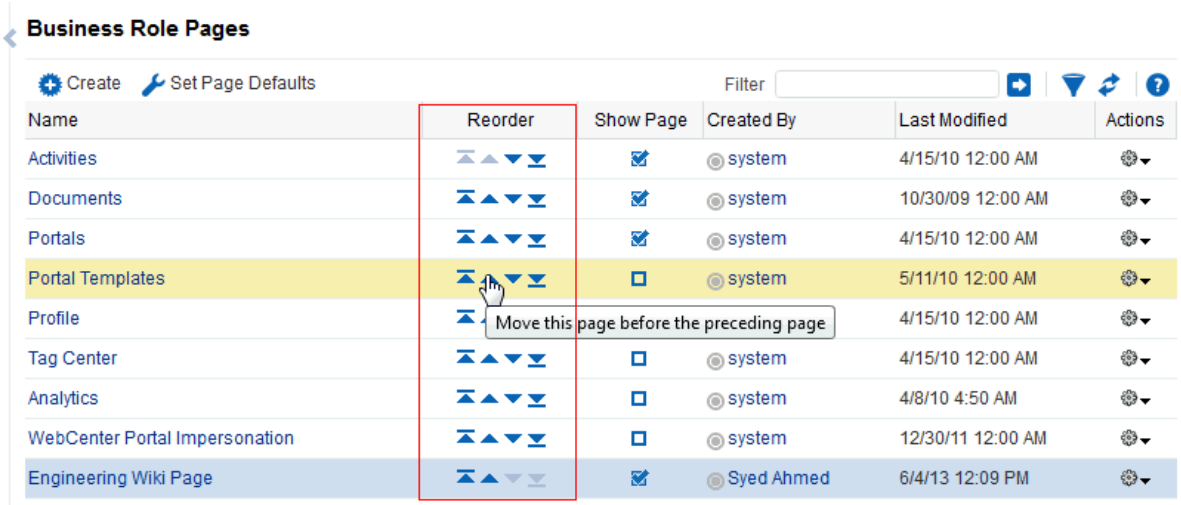
   You can also enter the following URL in your browser to navigate directly to the **Business Role Pages** page:

   `http://host:port/webcenter/portal/builder/administration/businessrolepages`

   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Select a business role page, and then click the arrows in the **Reorder** column to change the display order (Figure 51–10).

Figure 51–10 Reorder Icons on Business Role Pages



Alternatively, drag and drop pages into the desired order.

51.8 Editing a Business Role Page

Anyone granted the `Edit Page` permission on a business role page can edit that page. For these users, the editing process is the same as for regular pages (for more information, see the "Editing a Page" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal* and the "Editing a Personal Page" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

As the WebCenter Portal system administrator, you can also initiate an edit of a business role page from the **Administration** page.

To edit a business role page:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Business Role Pages**.

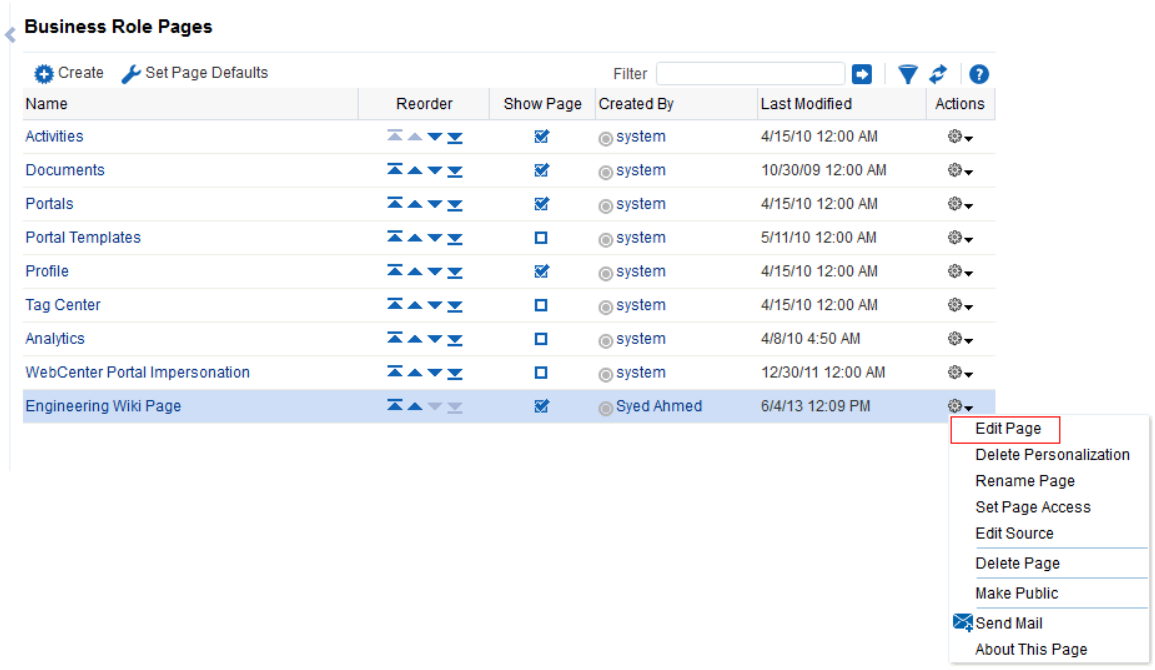
   You can also enter the following URL in your browser to navigate directly to the **Business Role Pages** page:

   `http://host:port/webcenter/portal/builder/administration/businessrolepages`

   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Select the page, click the **Actions** icon for the page you want to edit, and select **Edit Page** (Figure 51–11).

Figure 51–11 Edit Option on a Custom Business Role Page



The page opens in edit mode in Composer. For more information about editing a page in Composer, see the "Editing a Page" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

3. Edit the page, and click **Save** and then **Close** when you have finished.

51.9 Editing the Source of a Business Role Page

You can edit the source of a Business Role page without opening the page in Composer.

To edit the source of a Business Role page:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Business Role Pages**.

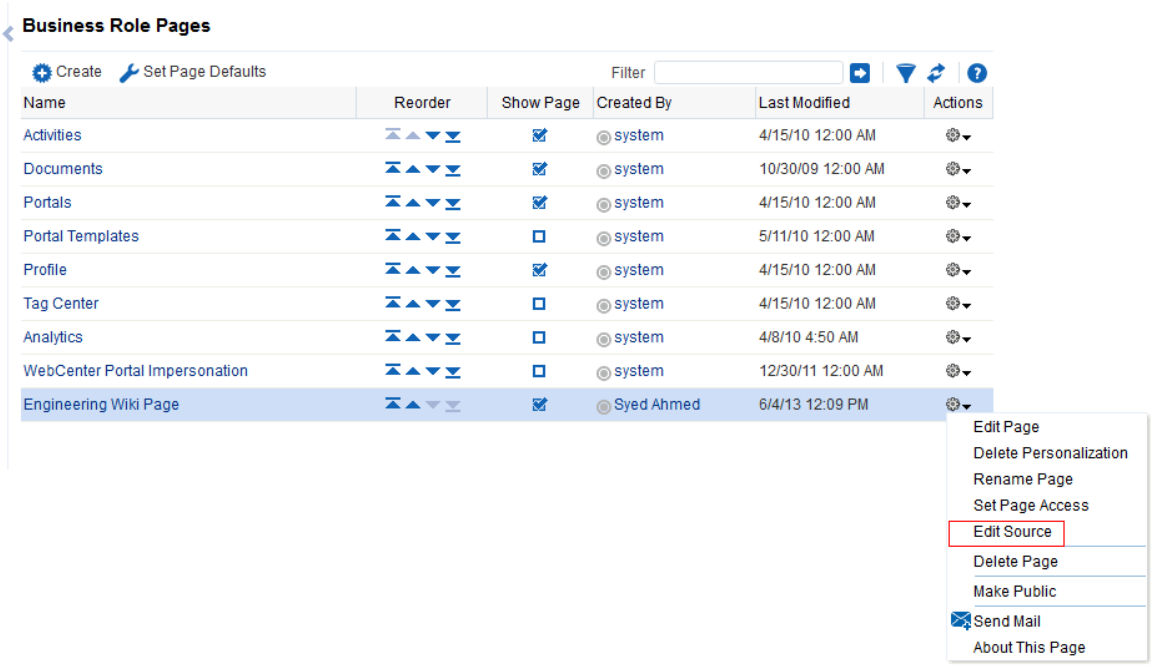
   You can also enter the following URL in your browser to navigate directly to the **Business Role Pages** page:

   `http://host:port/webcenter/portal/builder/administration/businessrolepages`

   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

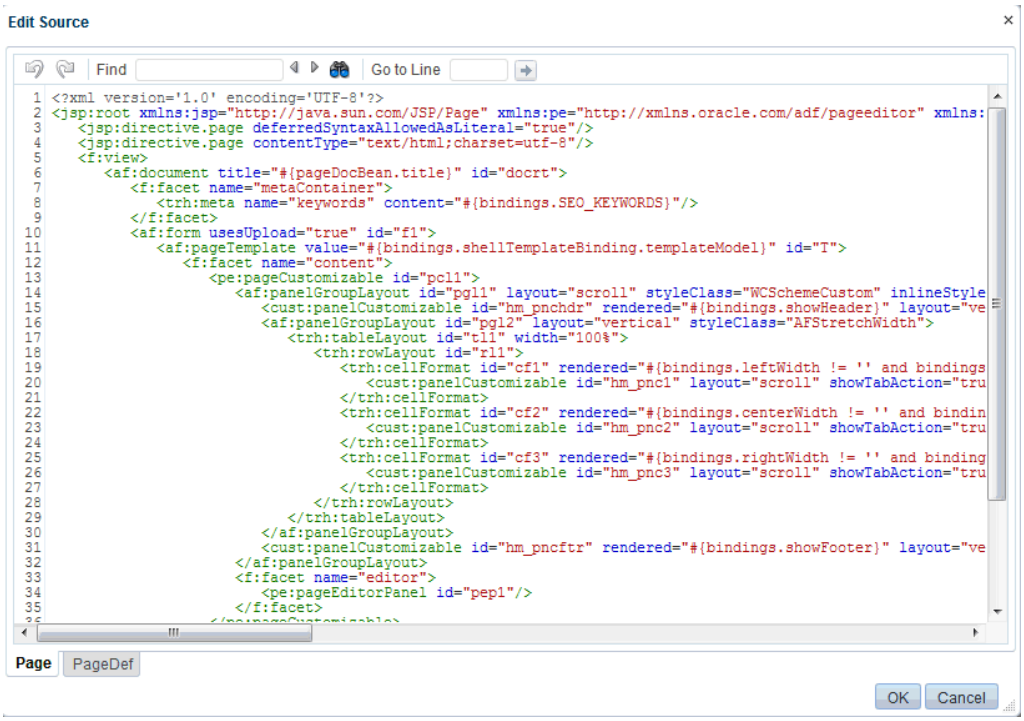
2. Click the **Actions** icon for the custom page whose source you want to edit, and select **Edit Source Page** (Figure 51–12).

Figure 51–12 Edit Source Option on a Custom Business Role Page



The Edit Source dialog opens (Figure 51–13).

Figure 51–13 Edit Source Dialog



3. Edit the page source, as desired.

   For more information about editing the source of a page, see the "Viewing and Modifying Page Source Code" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

4. Click **OK**.

51.10 Copying a Business Role Page

When you copy a business role page, you can save it as another business role page or as a personal page in your view of the Home portal. If you copy another business role page, you must set access on the new page because access permissions from the original page are not copied (for more information, see Section 51.4, "Specifying the Target Audience for a Business Role Page"). You cannot copy custom business role pages.

To copy a seeded business role page:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Business Role Pages**.

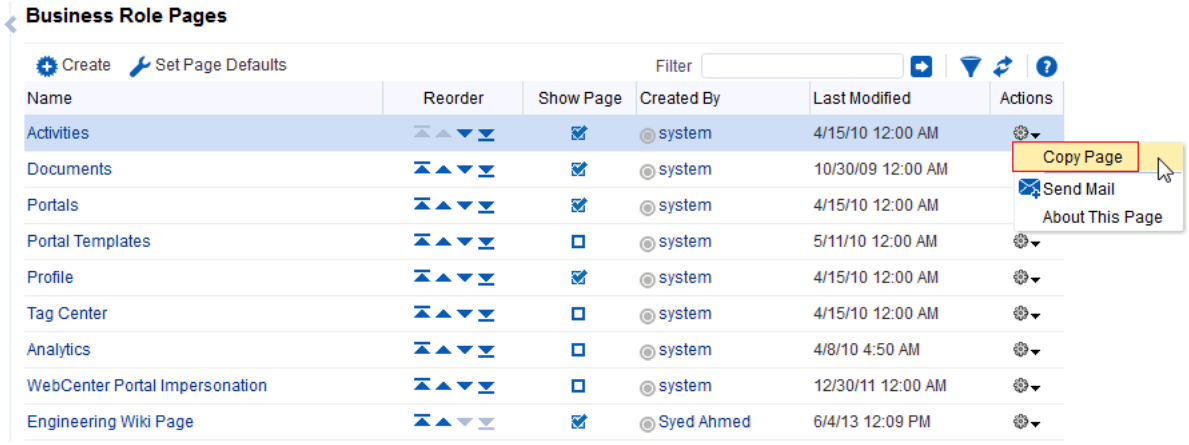
   You can also enter the following URL in your browser to navigate directly to the **Business Role Pages** page:

   `http://host:port/webcenter/portal/builder/administration/businessrolepages`

   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Select the page, click the **Actions** icon for the page you want to copy, and select **Copy Page** (Figure 51–14).

Figure 51–14 Copy Page Option on a Seeded Business Role Page



3. In the Copy Page dialog, enter a name for the new page (Figure 51–15).

Figure 51–15 Copy Page Dialog

4. Next to **Copy as**, specify whether the copy is a personal or business role page:
   - Select **Business Role Page** if you intend to expose the copy to a group of people with the same job role.
   - Select **Personal Page** if you intend to expose the copy only in your own view (that is, as a personal page in your view of the Home portal).
5. Click **OK**.

   The page opens in edit mode in Composer. For more information about editing a page in Composer, see the "Editing a Page" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.
6. Optionally, edit the page, and click **Save** when you have finished.

51.11 Removing All User Customizations from a Business Role Page

A control is available for removing all user customizations from a selected business role page. Using this control removes such personal changes as rearrangement, resizing, or collapsing of task flows. It does this in each user's personal view of the business role page.

To remove all user customizations from all views of a business role page:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Business Role Pages**.

   You can also enter the following URL in your browser to navigate directly to the **Business Role Pages** page:

`http://host:port/webcenter/portal/builder/administration/businessrolepages`

> **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. From the **Actions** menu next to the target page, select **Delete Personalization**.
3. In the resulting confirmation dialog, click **OK**.

   All user customizations added by users to their own views of the page are removed. That is, task flows are returned to their original positions and their original sizes; collapsed task flows are expanded; and so on.

51.12 Deleting a Custom Business Role Page

Anyone granted the `Delete Page` permission on a custom business role page can delete it. For these users, the process is the same as deleting regular pages (for more information, see the "Deleting a Page" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*. As the WebCenter Portal system administrator, you can also delete custom business role pages from the **Administration** page.

> **Note:** Seeded business role pages (see Table 51–1, " Seeded Business Role Pages") cannot be deleted, even by the system administrator.

After a custom business role page is removed from the WebCenter Portal, it cannot be recovered. Deleted pages are permanently removed, and users previously assigned that page no longer see it in their views of the Home portal.

To delete a custom business role page:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Business Role Pages**.

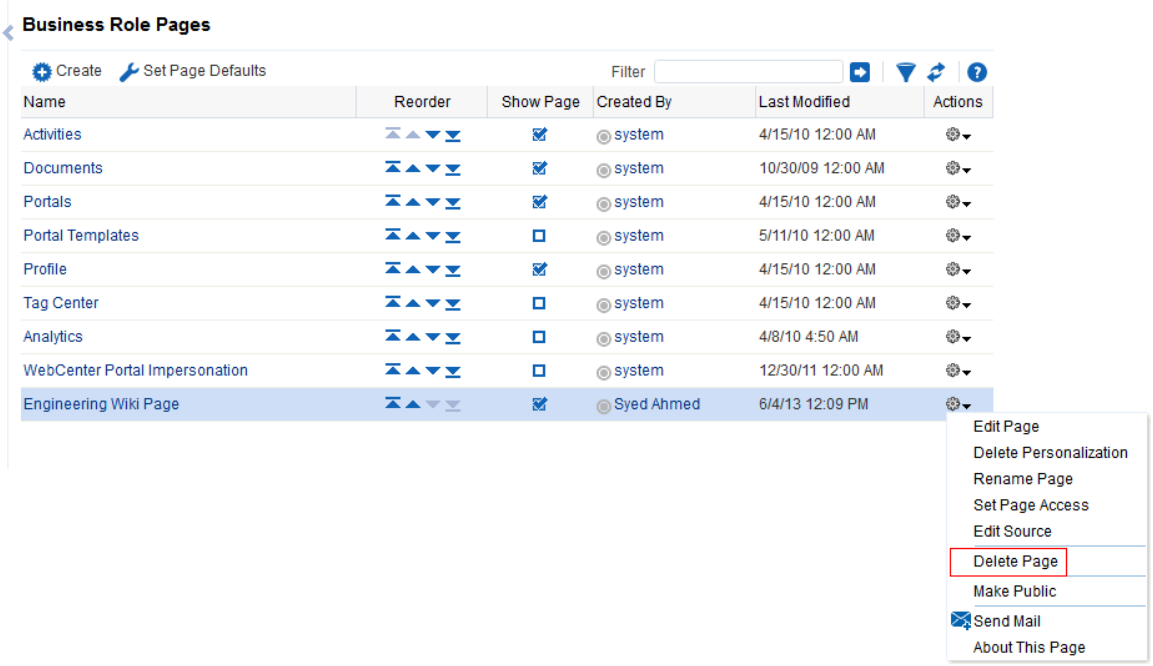
   You can also enter the following URL in your browser to navigate directly to the **Business Role Pages** page:

   `http://host:port/webcenter/portal/builder/administration/businessrolepages`

   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click the **Actions** icon for the page you want to delete, and select **Delete Page** (Figure 51–16).

Figure 51–16 Delete Page Option on a Custom Business Role Page



3. In the confirmation dialog, click **Delete**.

52 Managing Personal Pages

This chapter describes how to administer personal pages in WebCenter Portal. While individuals are primarily responsible for managing the content of their personal pages, WebCenter Portal system administrators also have access to all personal pages by default. System administrators may be required to clean up or manage personal data when owners experience difficulties with their personal pages or leave the organization.

This chapter includes the following topics:

Permissions: To perform the tasks in this chapter, you must have the WebCenter Portal `Administrator` role or a custom role that grants the following permission:

- `Portal Server-Manage Configuration`

Additionally, you need permissions to create and manage portals (`Portals-Create` and/or `Portals-Manage All`). For more information about permissions, see Section 49.3, "About Application Roles and Permissions."

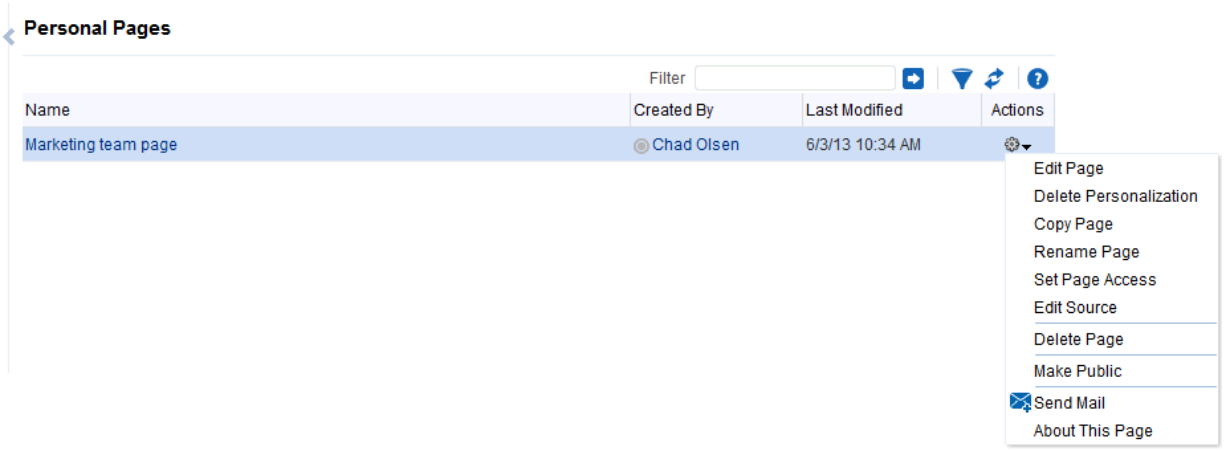
52.1 About Personal Page Administration

Personal pages are the pages users create in their personal views of the Home portal. As the WebCenter Portal system administrator, you have full access to all personal pages created by other users. Full access means you can edit, copy, rename, set access, delete, and perform other like actions on any user's personal pages.

System administrators can access everyone's personal pages from one, central place: the **Administration** page. The **Personal Pages** subpage provides access to a list of personal pages that includes information about who created the page and when it was last modified.

An **Actions** menu is associated with each listed page, providing access to options for editing in Composer, removing user customizations, copying, renaming, securing, editing the source, deleting, and making the personal page public (Figure 52–1).

Figure 52–1 Page Actions Menu on a Personal Page



Additional options include sending a mail message containing a link to the page and viewing information about the page.

52.2 Setting Application-Level Page Creation Defaults for Personal Pages

In addition to the page creation defaults authorized users can set for themselves (see the "Setting Page Creation Defaults for Personal Pages" section in *Oracle Fusion Middleware Using Oracle WebCenter Portal*), system administrators can set application-level page creation defaults for personal pages. After page creation defaults are configured, application-level page creation defaults affect the creation of all personal pages. This control (**Set Page Defaults**) is available on the **Business Role Pages** in WebCenter Portal Administration (for more information, see Section 51.2, "Setting Page Creation Defaults for Business Role Pages").

> **Note:** The page creation defaults that authorized users set for themselves through the **Personalize Pages** page in the Home portal override the application-level settings described in this chapter.

52.3 Changing Access Permissions on a Personal Page

As the system administrator, you are authorized to view and manage all personal pages. Page owners normally determine who can see their pages; however, as the system administrator, you have default access to all personal pages that other users create.

To change access permissions for a personal page:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Personal Pages**.

You can also enter the following URL in your browser to navigate directly to the **Personal Pages** page:

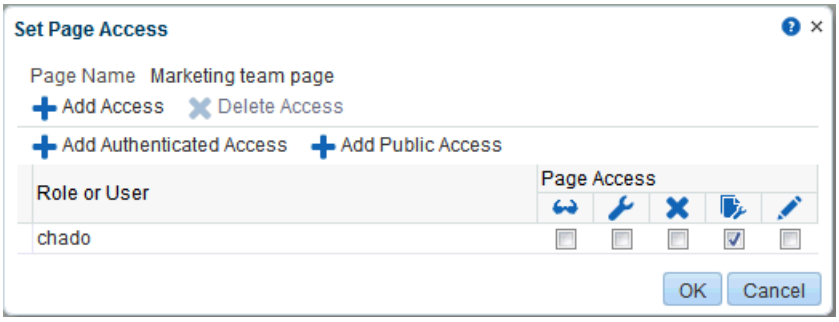
`http://host:port/webcenter/portal/builder/administration/personalpages`

> **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click the **Actions** icon for the page you want to secure, and select **Set Page Access** (see Figure 52–1).

   The Set Page Access dialog opens (Figure 52–2).

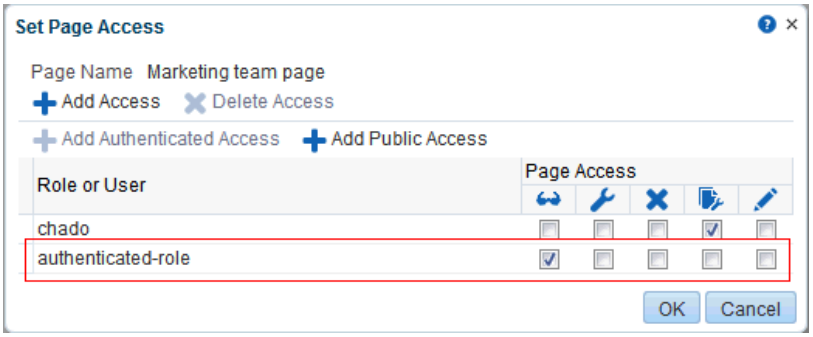
Figure 52–2 Set Page Access Dialog



3. To set access for authenticated users, click **Add Authenticated Access**.

   The role `authenticated-role` is added under **Role or User** (Figure 52–3). Use this to set access permissions for all users who are logged in to WebCenter Portal.

Figure 52–3 Authenticated Role Access

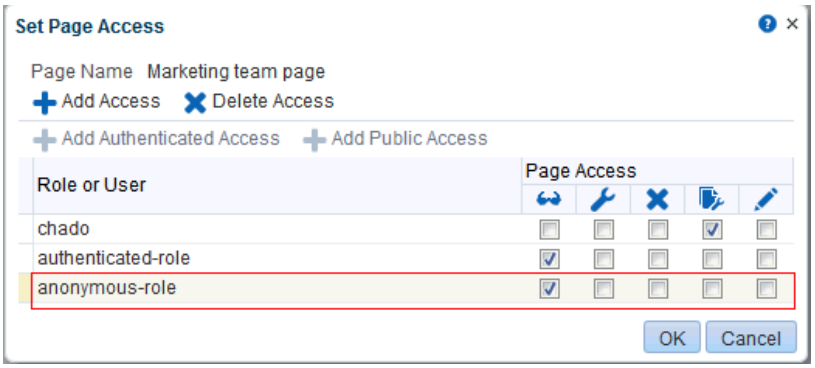


   Notice that the newly added role has only View access. Set other permissions for the user role appropriately.

4. To set access for public users, click **Add Public Access**.

   The role `anonymous-role` is added under **Role or User** (Figure 52–4). Use this to set access permissions for all users who are not logged in to WebCenter Portal.

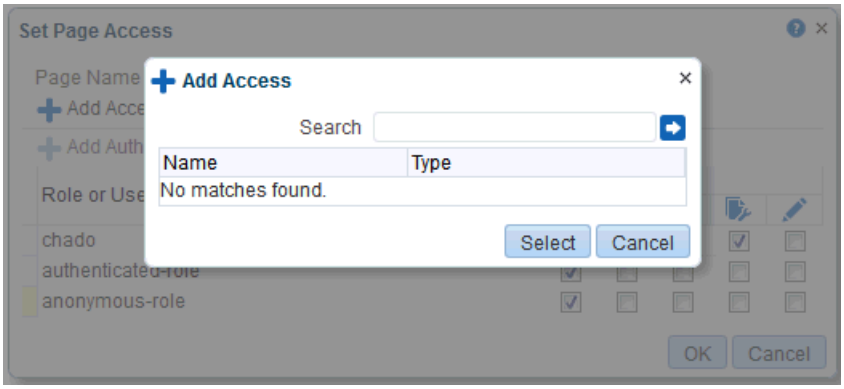
Figure 52–4 Anonymous Role Access



Notice that the newly added role has only View access. Set other permissions for the user role appropriately. You might want to leave the anonymous role with View access only.

5. To grant access to other users and roles, click **Add Access** to open the Add Access dialog (Figure 52–5).

Figure 52–5 Add Access Dialog



6. Identify the users who can access this page.

   Choose from all available users, groups, and application roles. Use the Search feature to search your identity store:

   a. In the **Search** field, enter two or more characters and click the **Search** icon. For tips on searching the identity store, see the "Searching for a User or Group in the Identity Store" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

      Tip: This search is not case sensitive.

      Users, groups, and roles matching your search criteria appear in the **Add Access** dialog.

   b. Select one or more names from the list.

      Press `Ctrl+Click` to select multiple users.

   c. Click **Select**.

The selected users and groups appear in the Set Page Access dialog. By default, users have the `View Page` permission on the page. Set other permissions appropriately.

7. To modify the permissions assigned to a current user or role, select one or more check boxes to grant page privileges:
   - **View Page**—The selected user or role can access the page for viewing, but cannot perform any actions on the page.
   - **Edit Page**—The selected user or role can edit the page. This includes adding, rearranging, and deleting content.
   - **Delete Page**—The selected user or role can delete the page.
   - **Perform All Page Actions**—The selected user or role has full access rights to the page. The user can edit the page, revise the page layout, set additional access privileges for other users, and all other page permissions.
   - **Personalize Page**—The selected user or role can personalize the page. Personalizations are changes made to a page in view mode. Such changes do not affect another user's view of the page.

     Tip: You can revoke privileges by taking the same steps and deselecting one or more privileges for a listed user or role.

   - To revoke access to the page, select the user or role and click **Delete Access**.
8. Click **OK**.

52.4 Preventing Users From Creating Personal Pages

You can revoke the application-level permission `Create-Pages` to prevent users from creating personal pages in the Home portal. The `Create-Pages` application-level permission applies only to creating personal pages; the creation of pages in a portal is controlled by the given portal's own security settings (for information about portal security settings, that is, portal-level users and roles, see the "Managing Members and Assigning Roles in a Portal" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

The process of revoking an application-level permission is described in the "Viewing and Editing Permissions of a Portal Role" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

52.5 Providing Navigation to Personal Pages

If you want to add a link to a personal page in your WebCenter Portal navigation, this process is described in detail in the "Adding Resources to a Navigation Model" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*. For detailed information about working with WebCenter Portal navigation, see the "Working with Navigations" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

52.6 Editing Personal Pages with Administrative Privileges

As the system administrator, you are authorized to view and modify any personal pages that users have created in their view of the Home portal. Individuals are primarily responsible for editing content on their personal pages, but, occasionally,

you may be required to edit such content. See also Section 52.7, "Editing the Source of a Personal Page."

To edit a personal page:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Personal Pages**.

   You can also enter the following URL in your browser to navigate directly to the **Personal Pages** page:

   `http://host:port/webcenter/portal/builder/administration/personalpages`

   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click the **Actions** icon for the page you want to edit, and select **Edit Page** (see Figure 52–1).

   The page opens in Composer.

   > **See Also:** To find out more about editing a page through Composer, see the "Editing a Page" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

3. Update the page, and click **Save** and then **Close** when you have finished.

52.7 Editing the Source of a Personal Page

You can edit the source of a personal page without opening the page in Composer.

To edit the source of a personal page:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Personal Pages**.

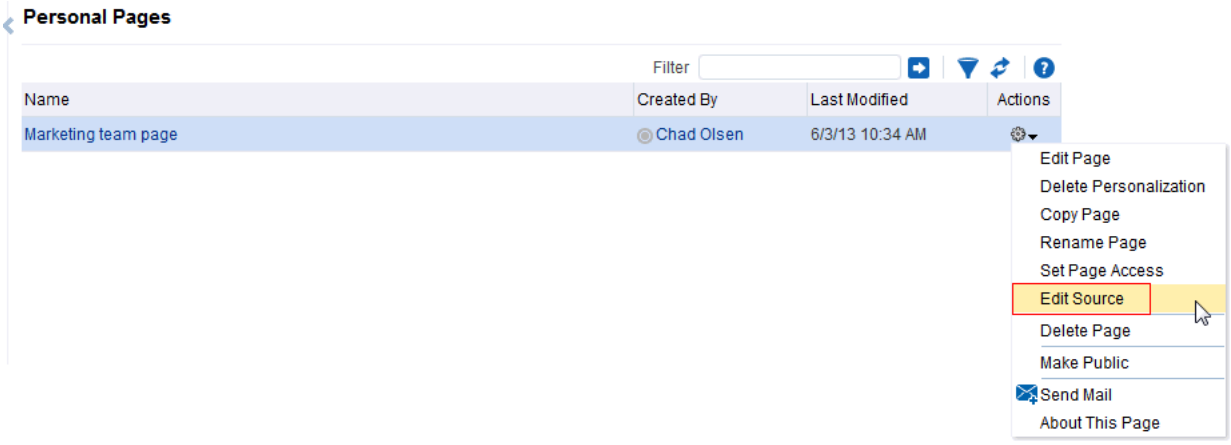
   You can also enter the following URL in your browser to navigate directly to the **Personal Pages** page:

   `http://host:port/webcenter/portal/builder/administration/personalpages`

   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

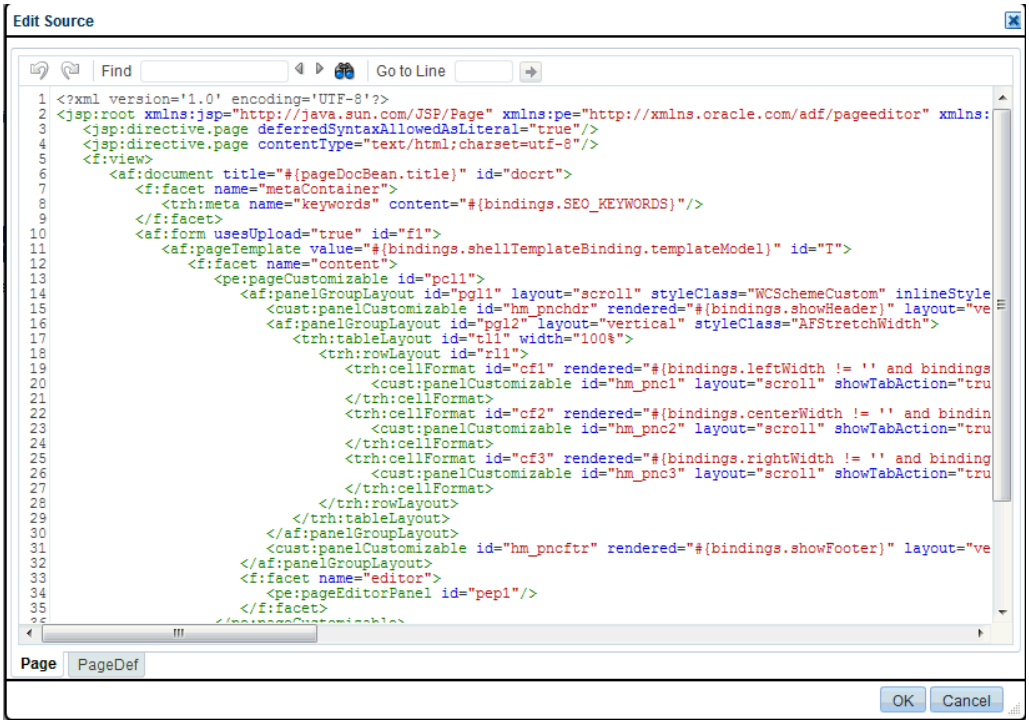
2. Click the **Actions** icon for the page whose source you want to edit, and select **Edit Source Page** (Figure 52–8).

Figure 52–6 Edit Source Option on Page Actions Menu



The Edit Source dialog opens (Figure 52–9).

Figure 52–7 Edit Source Dialog



3. Edit the page source, as desired.

   For more information about editing the source of a page, see the "Viewing and Modifying Page Source Code" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

4. Click **OK**.

52.8 Copying a Personal Page

As the system administrator, you are authorized to copy any page in the WebCenter Portal. This includes copying the personal pages created by other users. When you

copy a personal page as an administrator, you can save it as a business role page to be pushed to other users or as a personal page in your own view of the Home portal.

> **Tip:** If you create another business role page, you must set access on the new page because access permissions from the original page are not copied. For more information, see Section 51.4, "Specifying the Target Audience for a Business Role Page."

To copy a personal page:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Personal Pages**.

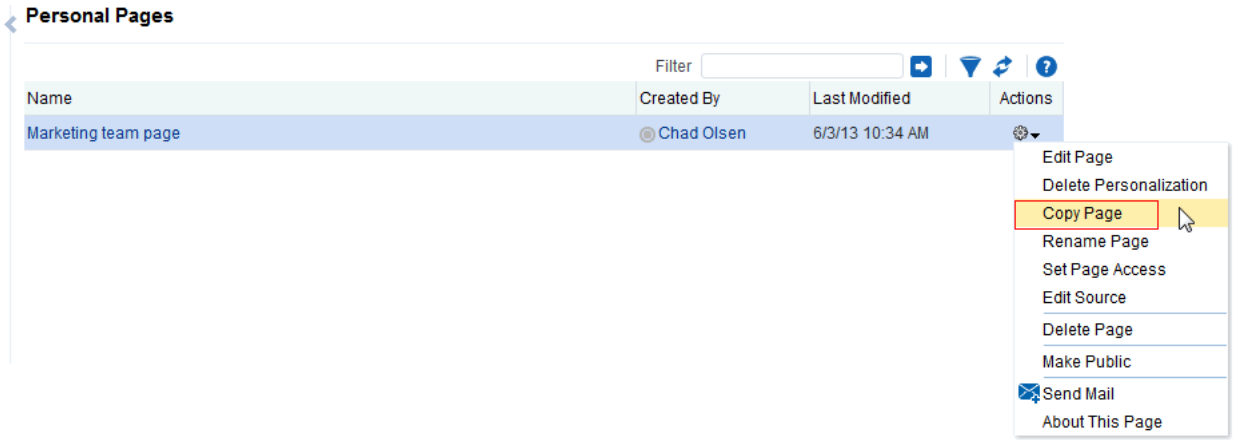
   You can also enter the following URL in your browser to navigate directly to the **Personal Pages** page:

   `http://host:port/webcenter/portal/builder/administration/personalpages`

   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

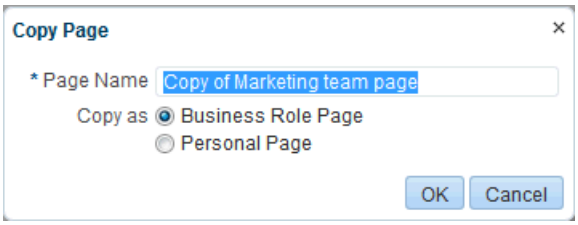
2. Click the **Actions** icon for the page you want to copy, and select **Copy Page** (Figure 52–8).

Figure 52–8 Copy Page Option on Page Actions Menu



The Copy Page dialog opens (Figure 52–9).

Figure 52–9 Copy Page Dialog



3. Enter a name for the new page.
4. Next to **Copy as**, specify whether the copy is one of your personal pages or a business role page:

- Select **Business Role Page** if you intend to make the page available to a group of people with the same job function or who are in the same enterprise group.
- Select **Personal Page** if you intend to expose the copy only in your own view.

To find out more about copying a page, see the "Copying a Page" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

5. Click **OK**.

   The new page opens in edit mode in Composer.

   See Also: To find out more about editing a page through Composer, see the "Editing a Page" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

6. Optionally, update the page, and click **Save** and then **Close** to exit Composer.

52.9 Removing All User Customizations from a Personal Page

A control is available for removing *all* user customizations from a selected personal page. Using this control removes such personal changes as rearrangement, resizing, or collapsing of task flows. The changes affect each user's personal view of the page.

To remove all user customizations from all views of a personal page:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Personal Pages**.

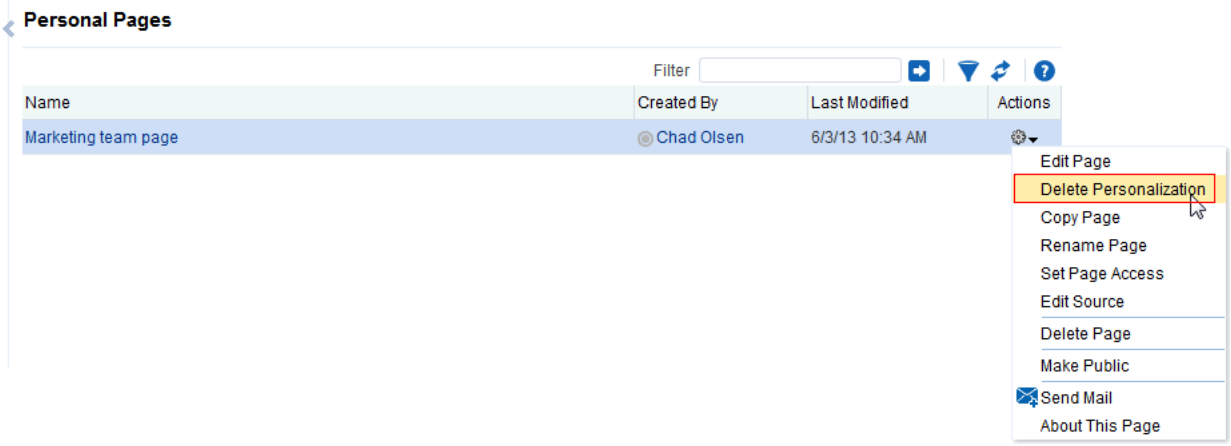
   You can also enter the following URL in your browser to navigate directly to the **Personal Pages** page:

   `http://host:port/webcenter/portal/builder/administration/personalpages`

   See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click the **Actions** icon next to the target page, select **Delete Personalization** (Figure 52–10).

Figure 52–10 Delete Personalization Option on Page Actions Menu



3. In the resulting dialog, click **OK**.

All user customizations added by users to their own views of the page are removed, that is, task flows are returned to their original positions and their original sizes; collapsed task flows are expanded; and so on.

52.10 Deleting a Personal Page Through the Portals Administration Page

In addition to having full access to the personal pages created by other users, a WebCenter Portal system administrator can also delete them, if required.

Note: After a personal page is deleted, it cannot be recovered.

To delete a personal page:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Personal Pages**.

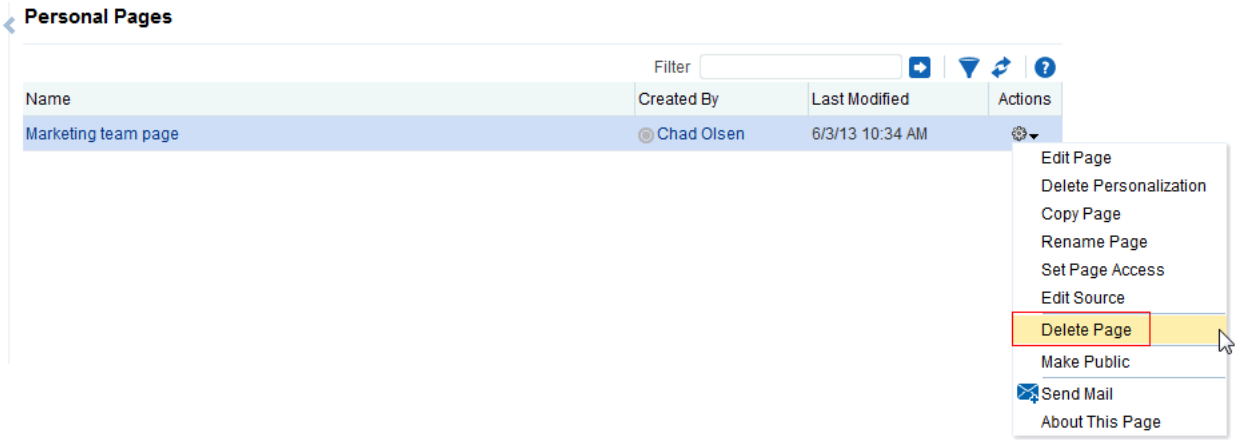
   You can also enter the following URL in your browser to navigate directly to the **Personal Pages** page:

   `http://host:port/webcenter/portal/builder/administration/personalpages`

   See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click the **Actions** icon for the page you want to delete, and select **Delete Page** (Figure 52–11).

Figure 52–11 Delete Page Option on Page Actions Menu



3. In the confirmation dialog, click **Delete**.

   To find out more about deleting a page, see the "Deleting a Page" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

53 Administering Device Settings

This chapter describes how to manage devices settings in WebCenter Portal. Device settings allow you to control how portals render on different kinds of devices including desktop browsers, smart phones, and tablets.

- Section 53.1, "About Device Settings"
- Section 53.2, "Creating and Managing Devices"
- Section 53.3, "Creating and Managing Device Groups"
- Section 53.4, "Managing Device and Device Group Life Cycles"
- Section 53.5, "Previewing Devices"
- Section 53.6, "Guidelines and Best Practices for Device Settings"
- Section 53.7, "Discovering Device Attributes: A Sample Task Flow"

Permissions: To perform the tasks in this chapter, you must have the WebCenter Portal `Administrator` role or a custom role that grants the following permission:

- `Portal Server-Manage Configuration`

For more information about permissions, see Section 49.3, "About Application Roles and Permissions."

53.1 About Device Settings

To successfully manage and administer device settings, you need to be familiar with the concepts described in this section:

- Section 53.1.1, "Introduction to Device Settings"
- Section 53.1.2, "What Are Devices?"
- Section 53.1.3, "What Are Device Groups?"
- Section 53.1.4, "Other Related Concepts"
- Section 53.1.5, "Basic Use Case: Adding Support for a New Device"
- Section 53.1.6, "Understanding How Device Settings are Applied"

53.1.1 Introduction to Device Settings

Enterprise portal users access portals from a range of devices, from smart phones to tablets to desktop browsers. Device settings and related features allow you to control

exactly how your portal pages render on different devices. As a system administrator, you may be asked to support a new type of device or to change or improve the way portal pages render on certain devices.

WebCenter Portal includes the capability to recognize which type of device a given request comes from, and to render the portal properly on that device. As a system administrator, you use device settings to modify or fine-tune this device recognition and to specify which page templates and skins to associate with specific devices or classes of devices. It is through device settings that you control exactly how those skins and templates are applied.

Out-of-the-box, WebCenter Portal provides several page templates that are designed to render well on general classes of devices, like smart phones, tablets, or desktop browsers. You can choose to use these templates as they are, modify them to suit your needs, or create new ones.

As a system administrator, consider using device settings when:

- You need to add rendering support for a new device or class of devices.
- You discover a problem with the way portal pages render on a device or class of devices.
- You find that portal developers have created device-specific pages that are not being detected and are not showing up on the targeted devices.

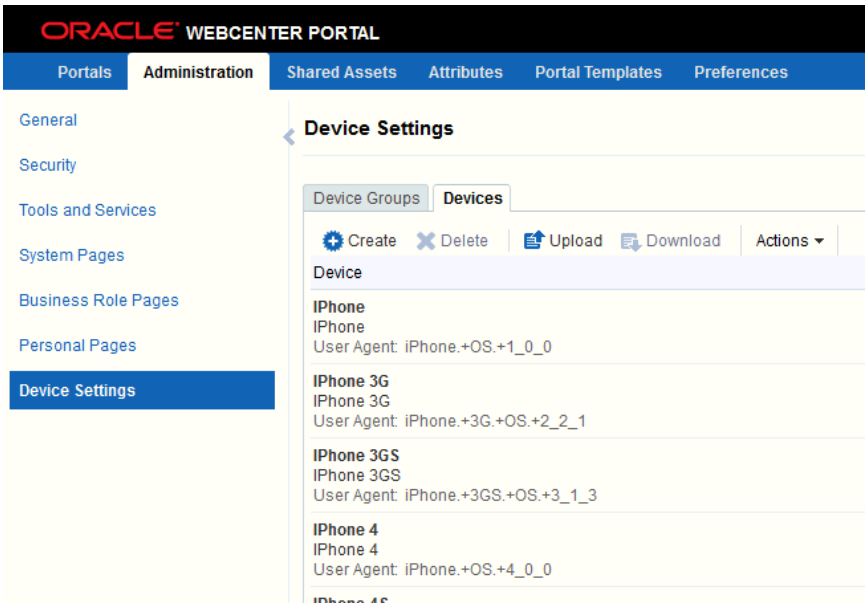
53.1.2 What Are Devices?

A *device* is a representation in WebCenter Portal of a physical device, like a smart phone or tablet, that users use to interact with a portal. Each time a portal page is requested, WebCenter Portal determines the type of device from which the request originated. This information enables the portal to decide what category of devices or "device group" the device is associated with. See Section 53.1.3, "What Are Device Groups?"

WebCenter Portal comes with a number of pre-configured devices out-of-the-box, such as iPhone, iPad, iPad mini, Samsung Galaxy Nexus, Samsung Galaxy Note 10.1, and others. You can also create new devices as needed.

Figure 53–1 shows some of the default devices listed in the **Administration** page.

Figure 53–1 List of Devices for Administrators



Each device has three primary characteristics: a name, a display name, and a user agent string:

- **Name** – A unique name for the device. One use of this name is that, for certain use cases, it can be located by a developer with an Expression Language expression.
- **Display Name** – This name will appear in the Portal Builder user interface.
- **User Agent** – A regular expression string that is used to identify the device from which a request originates. For example, an expression like `.*iPhone.+3G.+OS.+2_2.*` matches a variety of iPhone 3G versions.

> **Note:** The user agent string is a regular expression and conforms to the syntax specified by the Java platform (java.util.regex.Pattern). As such, certain special characters might need to be escaped if you want to match them. These characters include `[\^$.|?*+()` and, in some cases, curly brace characters `{}`. For example, a parenthesis must be escaped with "\", as in `\(iPhone; CPU iPhone OS 5_0 like Mac OS X\)`. For further guidance, a good reference on regular expression syntax is recommended.

As a system administrator, you can create new devices and manage existing ones. For example, you might need to modify the user agent string so to correctly identify a new version of a device. Or, you may need to create new devices as needed. For more information, see Section 53.2, "Creating and Managing Devices."

53.1.3 What Are Device Groups?

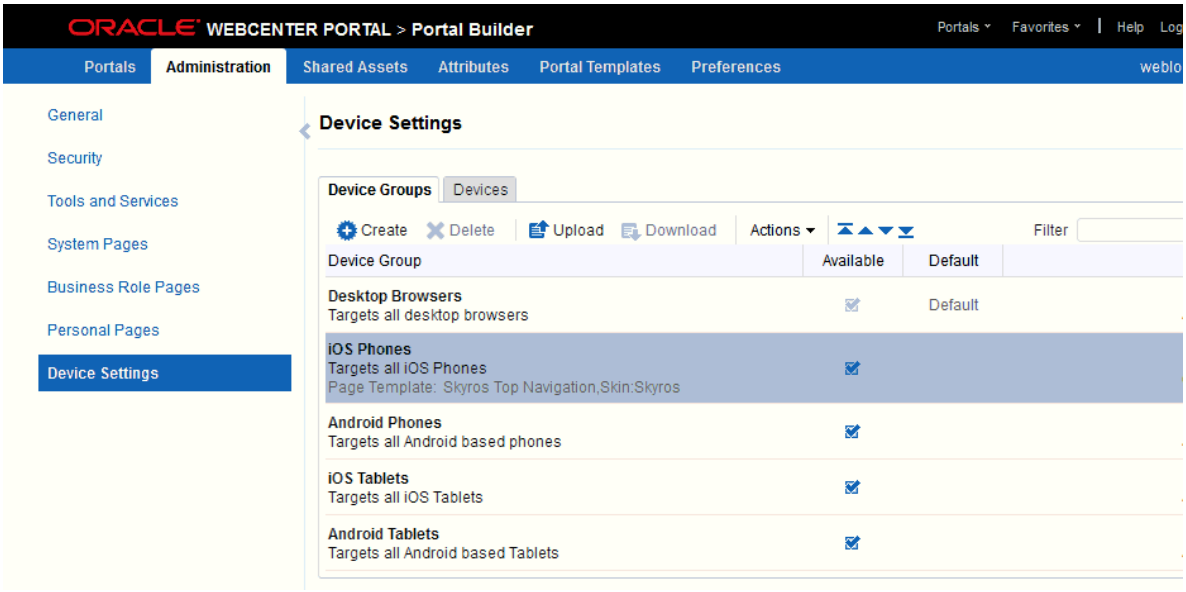
A *device group* represents a collection of devices that share similar display requirements. Out-of-the-box, WebCenter Portal comes with several pre-configured device groups: Desktop Browsers, iOS Phones, Android Phones, iOS Tablets, and Android Tablets.

Device groups are populated with appropriate devices. For example, the iOS Phones device group includes iPhone, iPhone 3G, iPhone 3GS, iPhone 4, iPhone 4S, and

others. As you create more devices, you can add them to existing groups, or create new groups as needed.

The advantage of device groups is that you do not have to configure display assets (page templates and skins) for each supported device. Rather, you can add multiple related devices to a group and specify the assets to be used by those devices. Figure 53–2 shows the Administration page for device groups. This page lets you create, edit, copy, upload, and perform other operations on device groups. For more information, see Section 53.3, "Creating and Managing Device Groups."

Figure 53–2 Device Group Administration Page



53.1.4 Other Related Concepts

The following features are related to device settings. They include the default device group, page variants, and the fallback page.

- **Default Device Group** – One device group is always specified as the default. Out-of-the-box, the default device group is Desktop Browsers. This means that, by default, all pages in a new portal are associated with the Desktop Browsers device group. If a request comes from an unrecognized device, the portal page is rendered according to the default device group settings.

  Note: The base page is always rendered on devices that belong to the default device group.

  The default device group is associated with the portal template feature (portal templates are templates on which new portals are based). Any portal created from a portal template automatically receives that template's default device group. Likewise, if you create a portal template from a portal, the default device group associated with that portal is placed into the template.

  For a discussion of how WebCenter Portal selects which device group to use, see Section 53.1.6, "Understanding How Device Settings are Applied."

- **Page Variant** – A *page variant* is an alternative view of an existing, or "base," page designed to be used with specific devices. The base page from which the variant is

derived and the page variant itself have the same URl, security settings, parameters, and so on; however, they are designed with specific rendering characteristics appropriate for the targeted device. When you create a page variant, you can specify a device group and page style with which to associate it.

Page variants have several uses. Suppose you find that one of your company intranet portal pages returns an error on a particular device. For example, such an error may occur when a page containing Flash video is rendered on an Apple device. In this case, you can create a variant of that page that will be used only when the page is requested for an Apple device, but not for others. In this case, the variant includes an image, perhaps, instead of the Flash video, and the error disappears.

Page variants are typically created by application specialists; however, only an administrator can create page variants for system pages. For example, you might want to create a login page variant suitable for a smart phone.

For information on creating page variants for system pages, see Section 50.2.2, "Creating a Page Variant of a System Page for Device Groups." For information on creating page variants for portal pages, see the "Creating a Page Variant for Devices" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

- **Fallback Page** – One other related concept is the *fallback page*. Whenever a page does not have a page variant, then the base page is rendered by default; however, you can override this behavior so that *no page* is displayed in this circumstance.

  Note: When a page's fallback behavior is set to **Display No Page**, any navigational links to that page (as defined in the navigation model) are hidden from view. In other words, any navigational links that would result in a "Page Not Available for Device" message are hidden from users.

  You can set fallback for individual pages or for all portal pages. For more information about fallback, see the "Setting the Device Group Behavior for a Portal When No Page Variant Exists" and "Setting the Device Group Behavior for a Page When No Page Variant Exists" sections in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

53.1.5 Basic Use Case: Adding Support for a New Device

Here is a use case to help you understand when you may need to work with device settings for the portal you administer.

Suppose a new mini-tablet is released with a different screen resolution and size than the currently supported tablet devices. In fact, a user discovers that the company intranet portal does not render properly on this device–there is a lot of white space and the company logo doesn't look right. You are asked to support this new device. The basic steps are:

1. Discover the user agent string that this device sends to the portal.
2. Create a new device that has a user agent string that can match the new device's user agent string. See Section 53.2.1, "Creating a New Device."
3. Create a new device group for all devices that share similar rendering characteristics to the new device. In this case, the device group would hold devices with similar display characteristics as the new tablet. See Section 53.3.1, "Creating

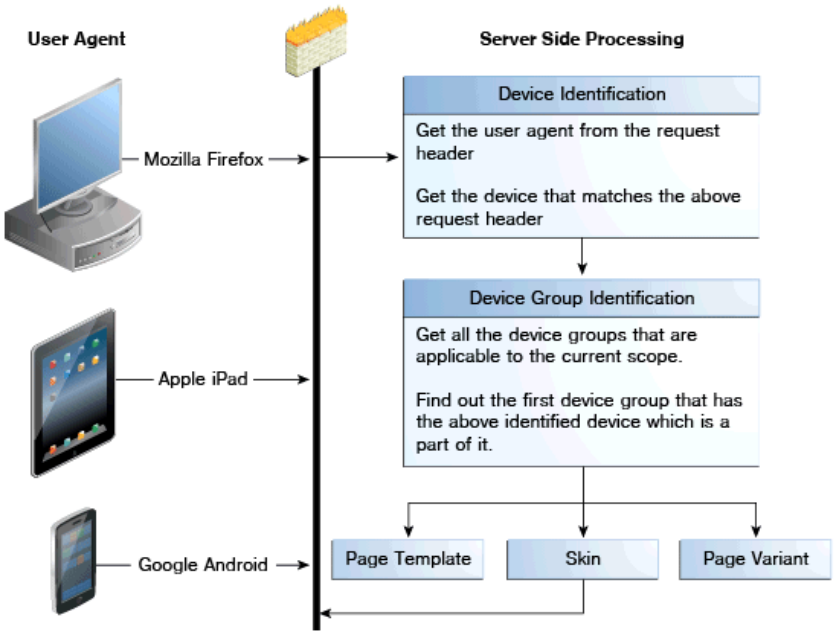
a Device Group."

4. Apply an appropriate skin and page template to the device group. If necessary, create new assets from scratch or by copy and modify existing ones.
5. Add the new device to the device group.
6. Test the portal on the new device to ensure it renders properly.
7. If similar mini-tablet devices are released, they can be added to the same group.

53.1.6 Understanding How Device Settings are Applied

Figure 53–3 illustrates the flow of how WebCenter Portal handles requests from multiple different devices.

Figure 53–3 How the Portal Handles Requests from Different Devices



As Figure 53–3 shows, when a request comes in to the server, the user agent string is examined in the header of the request. Next, WebCenter Portal looks for a device that matches that user agent (a regular expression string).

If multiple devices are defined whose user-agents can potentially map to the incoming user-agent, then the server tries to map the request to the *most appropriate* device. The most appropriate device is one whose user-agent has the maximal possible match.

When a device is identified, WebCenter Portal looks to see if it is in a device group. If it is in more than one group, the first one in the list of device groups for the portal is used. See also Section 53.3.6, "Ordering Device Groups."

If no device is identified, then WebCenter Portal assigns the "default device group" to the current request.

Finally, the appropriate skin, page template associated with the device group, and page variant (if one exists) are returned and the page renders on the device. If a page does not have a page variant, then the base page is rendered, by default; however, you can override this behavior so that no page is displayed in this circumstance. For more

information, see the "Setting the Default Page Fallback Behavior for Mobile Devices" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

53.2 Creating and Managing Devices

This section explains how to create and manage devices:

- Section 53.2.1, "Creating a New Device"
- Section 53.2.2, "Editing a Device"
- Section 53.2.3, "Copying a Device"
- Section 53.2.4, "Filtering the List of Devices"
- Section 53.2.5, "Deleting a Device"

Note: In some cases, users may be unable to view page variants after a device configuration is added or modified on the **Device Settings** page. The user must simply logout/login to the portal to clear the cache, and after that, the device will be recognized correctly.

53.2.1 Creating a New Device

To create a new device:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Device Settings**.

   You can also enter the following URL in your browser to navigate directly to the **Device Settings** page:

   `http://host:port/webcenter/portal/builder/administration/device`

   See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click the **Devices** tab.
3. Click **Create**.

   The Create Device page displays, containing three sections: **Device**, **Optional Attributes**, and **Additional Attributes**.

4. In the **Device** section, specify the following details:
   - **Name** - The name of the device. This name must be unique and cannot contain spaces. One use of this name is that it can be located with an Expression Language expression.
   - **Display Name** - Specify the display name of the device. This name must be unique and will appear in the Portal Builder user interface.
   - **User Agent** - Specify the user agent string. WebCenter Portal identifies a device by comparing the user agent string passed in the request header (comes from the user's device) and the string specified in this field. This parameter does not have to be a literal match with the request header. It is taken to be a regular expression, and you can enter any valid regular expression in this field.

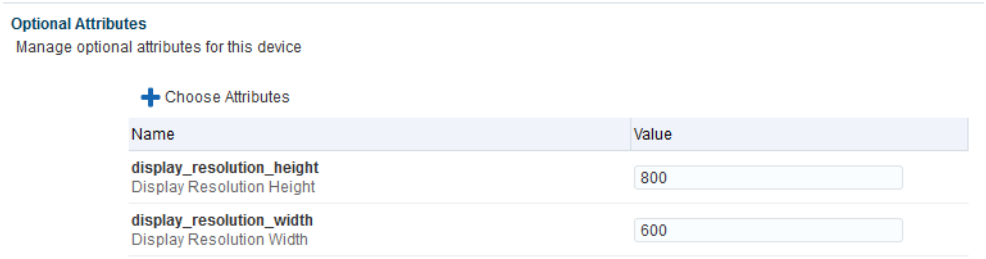
> **Note:** The user agent string is a regular expression and conforms to the syntax specified by the Java platform (java.util.regex.Pattern). As such, certain special characters might need to be escaped if you want to match them. These characters include `[\^$.|?*+()` and, in some cases, curly brace characters `{}`. For example, a parenthesis must be escaped with "\", as in `\(iPhone; CPU iPhone OS 5_0 like Mac OS X\)`. For further guidance, a good reference on regular expression syntax is recommended.

- **Description** - (Optional) Specify a description that helps to identify the purpose of the device.

5. Use the **Optional Attributes** section to manage attributes such as display resolution height and width. You can edit their default values as required, as shown in Figure 53–4.

> **Note:** Optional attributes do not affect the way portals are rendered on a device. They exist simply to provide a way to specify information about a device that may be useful to a page designer. Portal designers can use Expression Language to access the values of device attributes.

Figure 53–4 Specifying Optional Attributes



6. Optionally, add additional attributes. In the **Additional Attributes** section, click **Add Attribute** and specify a name and value.

> **Note:** Additional attributes do not affect the way portals are rendered on a device. They exist simply to provide a way to specify information about a device that may be useful to a page designer. Portal designers can use Expression Language to access the values of device attributes.

7. Click **Create** to create the device.

53.2.2 Editing a Device

To edit an existing device:

Note: In some cases, users may be unable to view page variants after a device configuration is added or modified on the Device Settings page. The user must simply logout/login to the portal to clear the cache, and after that, the device will be recognized correctly.

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Device Settings**.

   You can also enter the following URL in your browser to navigate directly to the **Device Settings** page:

   `http://host:port/webcenter/portal/builder/administration/device`

   See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Select the device you wish to edit.
3. From the **Actions** menu, select **Edit**.
4. Edit the device settings. For information about the device settings that can be edited, see Section 53.2.1, "Creating a New Device."
5. Click **Save**.

53.2.3 Copying a Device

Creating a copy of a device is useful when you want to:

- Create a backup of a device.
- Update a device while keeping the original in use.
- Use a built-in device as the starting point for creating a new device.

To copy a device group:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Device Settings**.

   You can also enter the following URL in your browser to navigate directly to the **Device Settings** page:

   `http://host:port/webcenter/portal/builder/administration/device`

   See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Select the device you wish to copy.
3. From the **Actions** menu, select **Copy**.
4. In the Copy dialog, specify the name, display name, user agent, and description of the device.
5. Click **OK**. The copied device appears in the Devices list.

53.2.4 Filtering the List of Devices

The Filter field lets you filter the list of devices shown in the Devices table. Filter searches on device group names, display names, descriptions, and user agents.

53.2.5 Deleting a Device

You can delete any device that you created or copied. You cannot delete any of the devices that are seeded out-of-the-box. To delete a device:

1. On the **Device Settings** administration page , click the **Devices** tab.
2. Select the device you wish to delete. (Use Ctrl-Click to select multiple devices).

> **Note:** You can only delete devices that you created or copied. You cannot delete the out-of-the-box devices that were provided with WebCenter Portal.

3. Click **Delete**.
4. Confirm your action in the Delete Device dialog.

53.3 Creating and Managing Device Groups

A device group represents a collection of devices that share similar display requirements. This section explains how to create and manage device groups to support:

- Section 53.3.1, "Creating a Device Group"
- Section 53.3.2, "Editing a Device Group"
- Section 53.3.3, "Copying a Device Group"
- Section 53.3.4, "Showing and Hiding Device Groups"
- Section 53.3.5, "Setting a Default Device Group"
- Section 53.3.6, "Ordering Device Groups"
- Section 53.3.7, "Filtering Device Groups"
- Section 53.3.8, "Deleting a Device Group"

See also Section 53.1.5, "Basic Use Case: Adding Support for a New Device."

> **Note:** In some cases, users may be unable to view page variants after a device configuration is added or modified on the Device Settings page. The user must simply logout/login to the portal to clear the cache, and after that, the device will be recognized correctly.

53.3.1 Creating a Device Group

To create a device group:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Device Settings**.

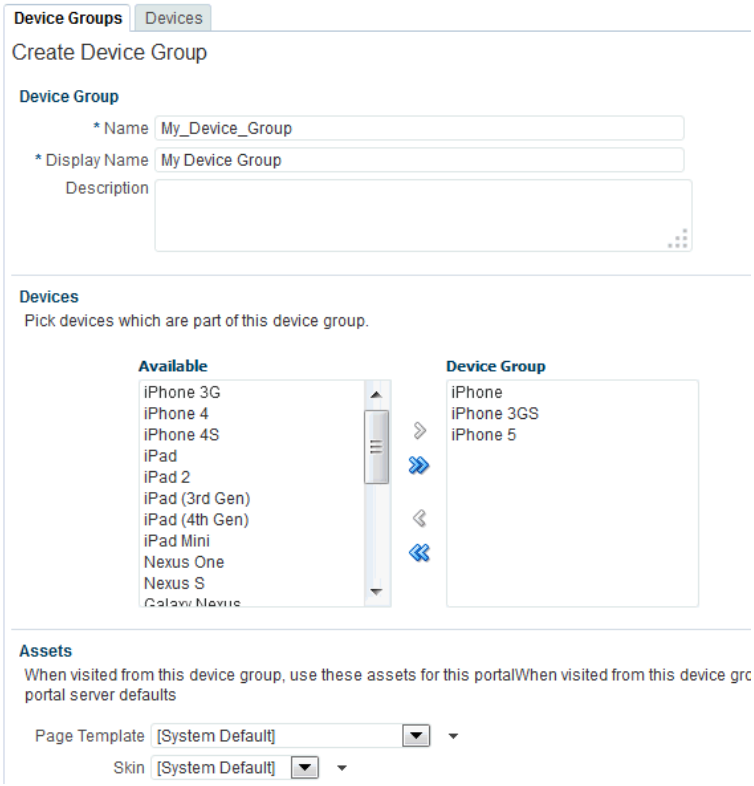
   You can also enter the following URL in your browser to navigate directly to the **Device Settings** page:

   `http://host:port/webcenter/portal/builder/administration/device`

   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click **Create** to open the **Create Device Group** page, as shown in Figure 53–5.

Figure 53–5 Creating a Device Group



3. Give the new device group a name and a display name. The name must be a unique name and is used internally. The display name is the name that is shown in Portal Builder. It also must be unique.
4. In the **Devices** section, use the arrows to move the available devices that you wish to add to the **Device Group** list.
5. In the **Assets** section, select the page template and skin that you want this device group to use.

> **Note:** Click the **Advanced Edit Options** arrow next to an asset, then **Expression Builder** to enter an EL expression in the Expression Editor. An EL allows the skin or template to be selected dynamically. If you need EL assistance, a developer can provide an EL expression; see the "Expression Language Expressions" appendix in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

6. To create the group, click **Create**.

53.3.2 Editing a Device Group

You can change the display name of a device group, edit the description that explains the purpose of the device group, and change the skin and/or template associated with

a device group. You cannot change the device group name that is internally used to identify it.

To edit the basic details of a device group:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Device Settings**.

   You can also enter the following URL in your browser to navigate directly to the **Device Settings** page:

   `http://host:port/webcenter/portal/builder/administration/device`

   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Select the device group you wish to edit, then click the **Actions** menu and select **Edit**.
3. On the **Edit Device Group** page, specify the required **Display Name** for the device group.
4. In the **Description** box, specify the purpose for which the device group has been created.
5. In the **Assets** section, select the page template and skin that you want this device group to use.

   > **Note:** Click the **Advanced Edit Options** arrow next to an asset, then **Expression Builder** to enter an EL expression in the Expression Editor. An EL allows the skin or template to be selected dynamically. If you need EL assistance, a developer can provide an EL expression; see the "Expression Language Expressions" appendix in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

6. Click **Save**.
7. Click **Close** to close the Edit Device Group page.

53.3.3 Copying a Device Group

You can create copies of device groups. This is useful when you want to:

- Create a backup of a device group.
- Update a device group while keeping the original in use.
- Use a built-in device group as the starting point for creating a new device group.

When you create a copy of a device group, the copy is marked as hidden regardless of the status of the original device group.

To copy a device group:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Device Settings**.

   You can also enter the following URL in your browser to navigate directly to the **Device Settings** page:

   `http://host:port/webcenter/portal/builder/administration/device`

See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Select the device group you wish to copy, then click the **Actions** menu and select **Copy**.
3. In the Copy dialog, specify the name, display name, and description of the device group (Figure 53–6).
4. Click **OK**.

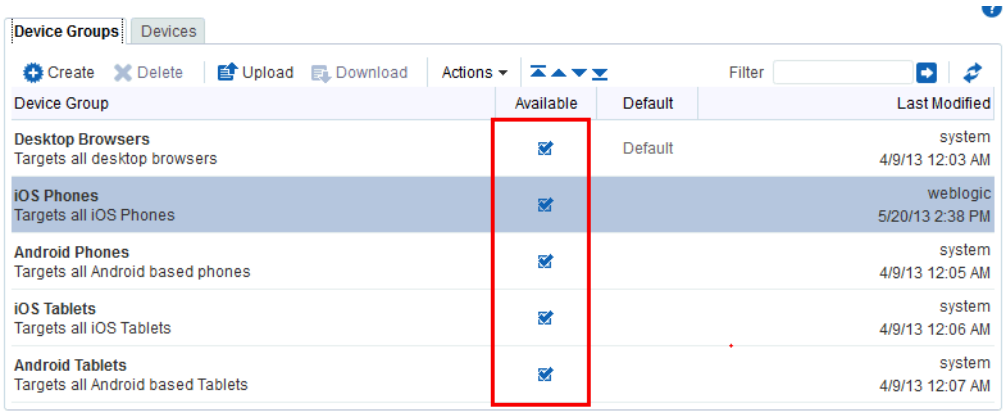
Figure 53–6 Copying a Device Group

53.3.4 Showing and Hiding Device Groups

All device groups, whether built-in or custom, can be marked as hidden or available. A check mark next to a device group's name indicates that the device group is available to use. An empty check box indicates that the device group is not available for use in portals (Figure 53–7). This setting also controls the available selections when you create page variants. If a device group is hidden, it does not show up as an option to use with a new page variant, and you can't create a page variant with that group. The show/hide settings are inherited from the Portal Administration settings; however, they can be overridden at the portal level by a portal moderator. See also the "Creating a Page Variant for Devices" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

When you create a device group, by default, it is marked as unavailable.

Figure 53–7 Available and Hidden Device Groups



To show or hide a device group:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Device Settings**.

You can also enter the following URL in your browser to navigate directly to the **Device Settings** page:

`http://host:port/webcenter/portal/builder/administration/device`

> **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. On the **Device Groups** tab, in the **Available** column, select or deselect checkbox in the **Available** column to show or hide the device group.

53.3.5 Setting a Default Device Group

The built-in device group named Desktop Browsers is the default device group in WebCenter Portal. All new pages that you create are automatically associated with the default device group.

On the **Device Groups** tab, `Default` appears next to the device group that is set as default.

To set a device group as default:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Device Settings**.

   You can also enter the following URL in your browser to navigate directly to the **Device Settings** page:

   `http://host:port/webcenter/portal/builder/administration/device`

   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click the **Device Groups** tab.
3. On the **Device Groups** tab, select the device group that you want to specify as default, then click the **Actions** menu and select **Set as Default**.

   Notice that `Default` now appears next to the selected device group.

53.3.6 Ordering Device Groups

When a user accesses WebCenter Portal using a device, portals are rendered using the assets like page template and skin associated with the device group to which that device belongs. However, a device may be associated with multiple device groups. In such cases, the ordering of the device groups in the **Device Groups** tab determines the precedence of device groups.

To define the order of the device groups:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Device Settings**.

   You can also enter the following URL in your browser to navigate directly to the **Device Settings** page:

   `http://host:port/webcenter/portal/builder/administration/device`

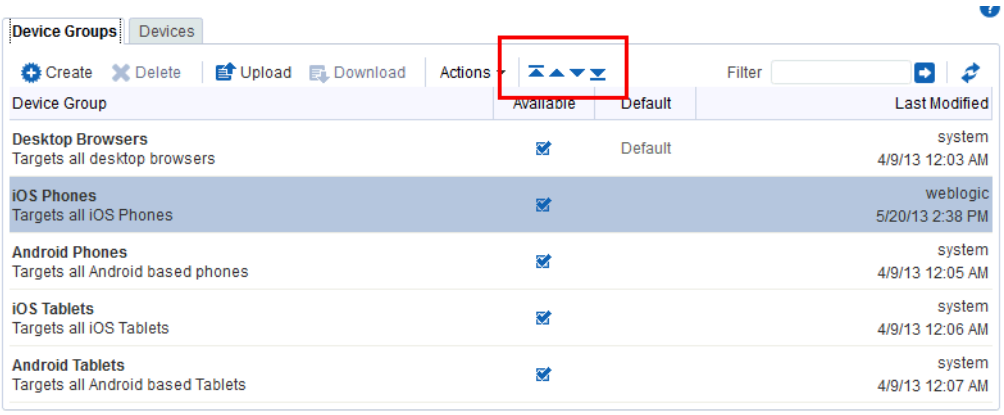
   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click the **Device Groups** tab.

3. Use the ordering icons to define the order of the device groups:
   - **Move to top**: Click to move the selected device group to the top in the list of device groups displayed.

     This implies that if a device belongs to more than one device group, then the topmost device group must take precedence.
   - **Move up**: Click to move the selected device group one level up in the list of device groups displayed.
   - **Move down**: Click to move the selected device group one level down in the list of device groups displayed.
   - **Move to bottom**: Click to move the selected device group to the end in the list of device groups displayed.

Figure 53–8 Reordering Device Groups



53.3.7 Filtering Device Groups

The **Filter** field lets you filter the list of device groups shown in the **Device Group** table. Filtering searches on device group names, display names, and descriptions.

53.3.8 Deleting a Device Group

If you no longer require a device group, you may want to delete it. However, you can delete only the custom device groups, and not the built-in device groups.

> **Note:** If a device group is deleted, page variants associated with that device group will still exist (they are not deleted). For important guidelines related to deleting a device group, see Section 53.6, "Guidelines and Best Practices for Device Settings."

To delete a device group:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Device Settings**.

   You can also enter the following URL in your browser to navigate directly to the **Device Settings** page:

   `http://host:port/webcenter/portal/builder/administration/device`

> **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click the **Device Groups** tab.
3. Select the device group that you want to delete, then click **Delete**.
4. In the Delete Device Group dialog, click **Delete**.

53.4 Managing Device and Device Group Life Cycles

You can download device groups and devices to a file, and then upload them to another WebCenter Portal instance. For example, if you want to move your device groups from a staging to a production server, use the life cycle mechanism described in this section:

- Section 53.4.1, "Downloading a Device Group or Device"
- Section 53.4.2, "Uploading a Device Group or Device"

Note:

- You can only download device groups or devices that you have copied or created. You cannot download any of the out-of-the-box device groups.
- When you upload or download a device group, all artifacts associated with that device group are included, including any devices associated with that group. For example, suppose you create a new device and add it to a group, then download the group. When you upload that group to another server, that new device is automatically added to the list of devices.

53.4.1 Downloading a Device Group or Device

To download a device group or a device to a file:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Device Settings**.

   You can also enter the following URL in your browser to navigate directly to the **Device Settings** page:

   `http://host:port/webcenter/portal/builder/administration/device`

   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click the **Device Groups** or **Devices** tab.
3. Select the device group or device you wish to download. You can use Ctrl+Click to select multiple rows.
4. Click **Download**.
5. In the Download dialog, in the **Archive File Name** field, enter a name for the archive file.
6. Select:

- **Save to My Computer** to save the archive file to your local file system. When you click the **Download** button you are prompted for the location on the file system where you want to save the file.
- **Save to WebCenter Portal Server** to save the archive file to the file system of the server. In the **Path** field, enter the location on the server where you want to save the archive file. See also the "Downloading an Asset" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

53.4.2 Uploading a Device Group or Device

To upload a previously downloaded device or device group to the portal:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Device Settings**.

   You can also enter the following URL in your browser to navigate directly to the **Device Settings** page:

   `http://host:port/webcenter/portal/builder/administration/device`

   See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Select the **Device Groups** or **Devices** tab.
3. Click **Upload**.
4. Use the Upload Devices/Device Groups dialog to locate the `.ear` file on your system.
5. Click **Upload**.

53.5 Previewing Devices

WebCenter Portal includes a preview feature that lets you preview how pages and page variants will render on a particular device. For more information, see the "Previewing a Mobile Device Variant of a Page" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

53.6 Guidelines and Best Practices for Device Settings

This section discusses best practices for working with Device Settings.

Avoid Changing the Default Device Group for a Production Portal

Changing the default device group in a production portal can lead to unexpected behaviors. It is best to avoid changing the default device group after your portal is in production.

Avoid Deleting a Custom Device Group for a Production Portal

If you delete a custom device group for a production or portals, the server does not warn you that existing portals use that device group, leading to incorrect page renderings in some cases.

If You Accidentally Delete a Device Group

You can create another device group with the same device group name as the one that was deleted. When a device group is deleted, the page variants are not removed from

the system. These page variants are associated using the name of the device group. Recreating a device group with the same will bring back all those pages.

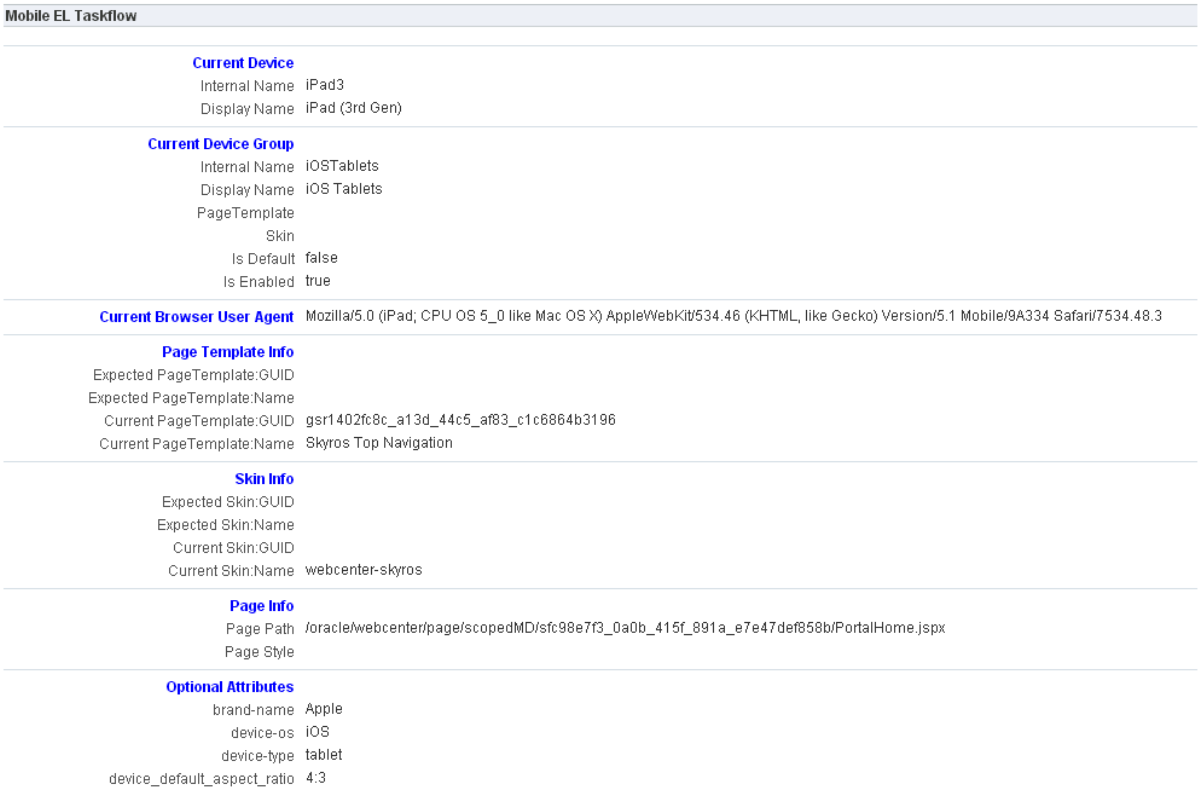
If You Need Information About the Requesting Device

In some cases, it is useful to obtain information about the device used to access the portal and discover which device settings the portal is mapping the device to. For your convenience, Section 53.7, "Discovering Device Attributes: A Sample Task Flow," lists code that a developer can use to create a task flow that echoes back this device information.

53.7 Discovering Device Attributes: A Sample Task Flow

Expression Language expressions can be used to return device attributes. Sample code that can be used for this purpose is presented in the appendix "EL Expresssions Related to Device Settings" in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*. A developer can use this code to create a task flow that returns device information that can be useful in troubleshooting a problem with the way a portal renders on a given device. Figure 53–9 shows the output from a task flow created with this sample code.

Figure 53–9 Output from Sample Task Flow



54

Customizing Task Flows Across Portals

This chapter describes how to use the **Task Flow Editor** system page to customize task flows for use by all portals in WebCenter Portal.

> **Note:** Task flow customization is also possible at design time through Oracle JDeveloper. The process differs significantly from the runtime procedure discussed in this chapter. For more information, see the "Customizing Task Flows" chapter in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*

This chapter includes the following topics:

- Section 54.1, "About Task Flow Customization at the Application Level"
- Section 54.2, "Customizing Task Flows at the Application Level"
- Section 54.3, "Removing Task Flow Customizations"

> **Permissions:** To perform the tasks in this chapter, you must have the WebCenter Portal `Administrator` role or a custom role that grants the following permission:
>
> - `Portal Server-Manage Configuration`
>
> Additionally, you need permissions to create and manage portals (`Portals-Create` and/or `Portals-Manage All`). For more information about permissions, see Section 49.3, "About Application Roles and Permissions."

54.1 About Task Flow Customization at the Application Level

Task flow customization provides a means of configuring a particular task flow in a way that all instances of that task flow within the current scope are affected. For example, you can add a link or icon to a task flow that requires it for all portals.

The task flow customization feature is available exclusively on the **Task Flow Editor** system page. The **Task Flow Editor** system page is available for both the application (all portals) and for individual portals:

- To change all instances of a given task flow across all portals (including the Home portal), customize the task flow on the application-level **Task Flow Editor** system page, as described in this chapter.

- To change only those instances exposed in a given portal, the portal moderator can customize the task flow on the portal-level **Task Flow Editor** system page. See the "Customizing Task Flows for a Portal" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

> **Note:** When you customize a task flow element at the application level, and another user customizes the same task flow element at the portal level, the portal-level customization take precedence in that portal.

The **Task Flow Editor** system page is provided to enable customization of any out-of-the-box task flow. Custom task flows that are created through the **Assets** or **Shared Assets** page cannot be customized in this way.

System pages have a Restore Default feature that enables authorized users to remove all page customizations and restore a system page to its out-of-the-box state. It is important to note that Restore Default does not also restore customized task flows to their default states. A separate control, Reset Task Flow, is available to remove task flow customizations.

> **See Also:** For information about the Restore Default and Reset Task Flow features for system pages, see Section 50.4, "Removing All Page Customizations from a System Page" and Section 54.3, "Removing Task Flow Customizations," respectively.

54.2 Customizing Task Flows at the Application Level

This section describes how to perform task flow customizations for WebCenter Portal, at the application level.

> **Note:** When you customize a task flow element at the application level, then another user customizes the same task flow element at the portal level, the portal-level customization take precedence in that portal.

To perform application-wide task flow customizations through the **Task Flow Editor** system page:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **System Pages**.

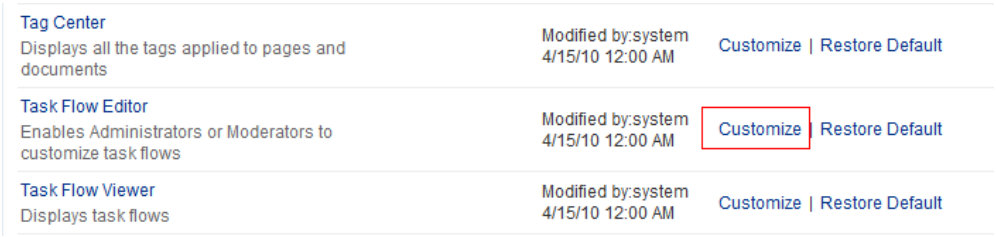
   You can also enter the following URL in your browser to navigate directly to the **System Pages** page:

   `http://host:port/webcenter/portal/builder/administration/systempages`

   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click the **Customize** link next to the **Task Flow Editor** system page (Figure 54–1) to open it in the page editor (Composer).

Figure 54–1 Customize Link Next to the Task Flow Editor System Page

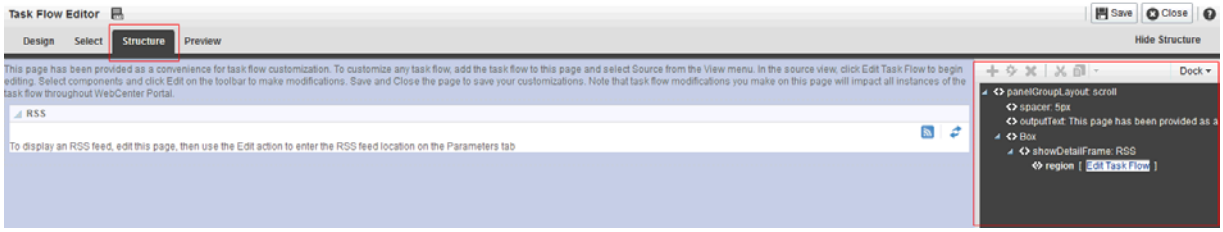


3. Add a task flow that you want to edit to the **Task Flow Editor** page.

   The method for adding a task flow to a page is that same as for any other component in the resource catalog. For more information, see the "Adding a Component to a Page" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

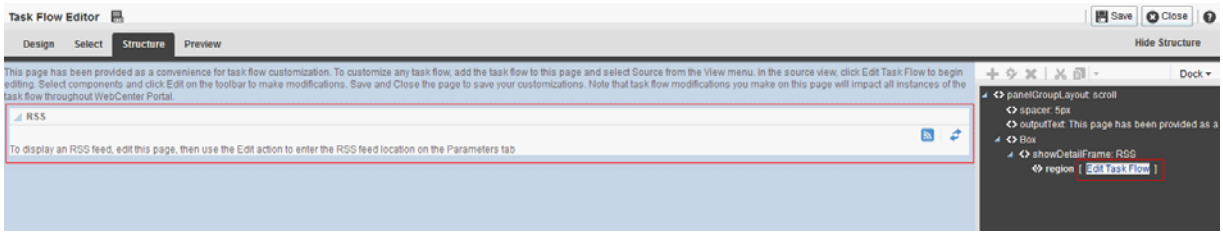
4. Click **Structure** to view the page source (Figure 54–2).

Figure 54–2 Structure View of Task Flow Editor System Page



5. Click the **Edit Task Flow** link next to the task flow you want to customize (Figure 54–3).

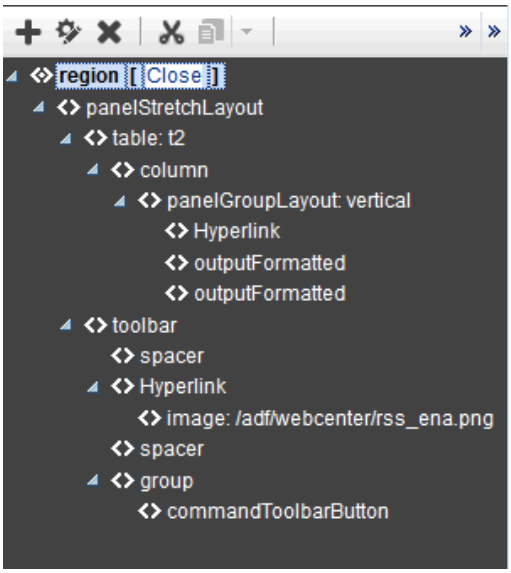
Figure 54–3 Edit Task Flow Link Next to a Region Tag



6. In the Confirm Task Flow Edit dialog, click **Edit**.

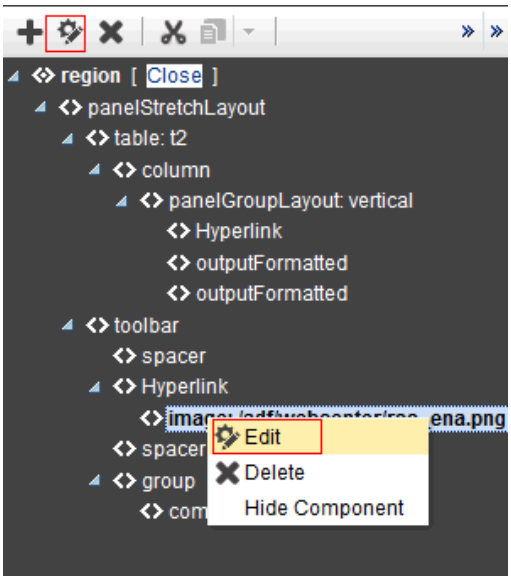
   Structure view zooms into the source code hierarchy of the task flow being edited (Figure 54–4)

Figure 54–4 Zoomed-In View of Task Flow



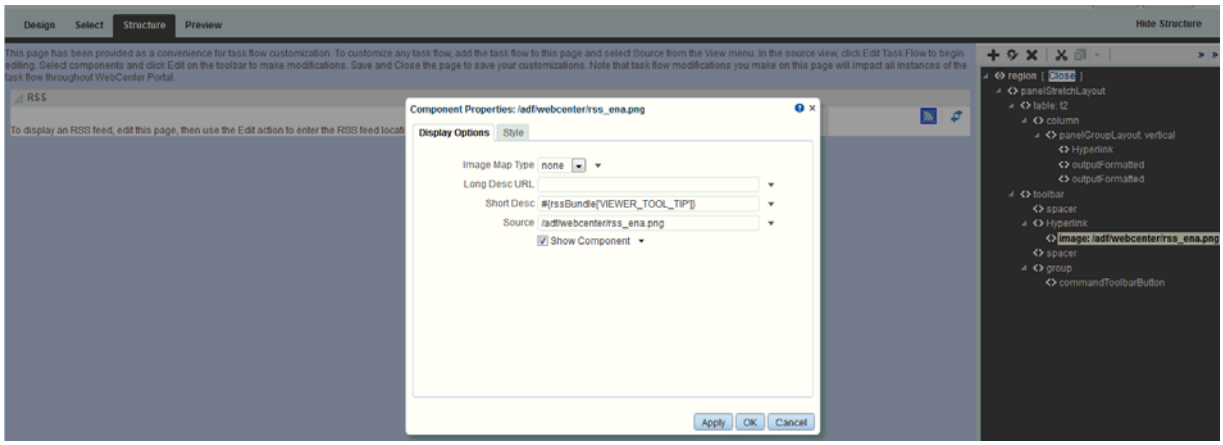
7. Set the properties of a task flow element by clicking it in the Task Flow Editor, then click the **Show the properties of *region*** icon. Alternatively, right-click the `region` and select **Edit** (Figure 54–5).

Figure 54–5 Selected Task Flow Element on a Page in Structure View



The Component Properties dialog opens (Figure 54–6).

Figure 54–6 Component Properties Dialog



8. Make your changes to the element's properties.

   For more information about editing components, see the "Adding and Editing Resource Catalog Components on a Page" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*. For more information about components, see the "WebCenter Portal Components" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

   Note: Remember that changes to one element affect all like elements in the task flow within the current scope. For example, a change to the font used on a folder name affects all folder names within the scope and not just the selected instance.

9. Click **Apply** to view the effect of your changes; click **OK** to save your changes and exit the dialog.

   Every instance of the customized task flow within the current scope renders with your customizations.

10. Click **Save** then **Close** to exit Composer.

54.3 Removing Task Flow Customizations

You can remove all customizations made to seeded task flows in WebCenter Portal.

Note: This procedure does not apply to task flows created at runtime. That is, task flows created through the **Assets** or **Shared Assets** pages. Changes made to a task flow created at runtime are base edits rather than layered customizations; therefore, when you click **Reset Task Flow**, there are no customization layers to remove.

To remove task flow customizations made at the application level:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **System Pages**.

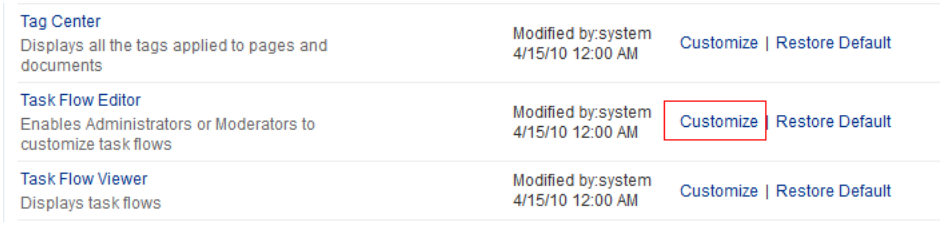
   You can also enter the following URL in your browser to navigate directly to the **System Pages** page:

`http://host:port/webcenter/portal/builder/administration/systempages`

See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

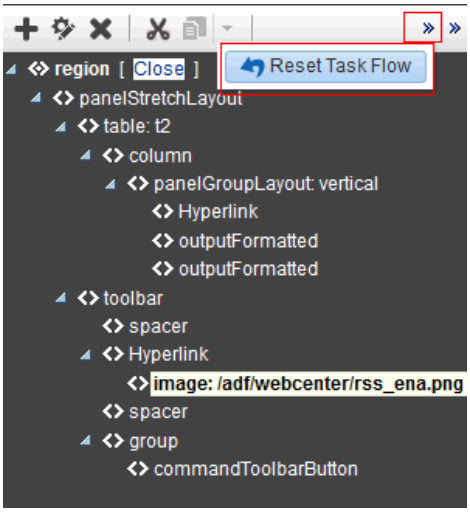
2. Click the **Customize** link next to the **Task Flow Editor** system page (Figure 54–7) to open it in page edit mode.

Figure 54–7 Customize Link Next to a System Page



3. Click **Structure** and select the customized task flow.
4. In the Confirm Task Flow Edit dialog, click **Edit Task Flow**.
5. Click **Reset Task Flow**.

Figure 54–8 Reset Task Flow Option for a Selected Element



6. In the Reset Task Flow dialog, click **Reset Task Flow** to confirm the action.

55

Working with Global Attributes Across Portals

This chapter describes how a system administrator can manage global attributes, which can be used by any portal in WebCenter Portal.

This chapter includes the following topics:

- Section 55.1, "About Global Attributes"
- Section 55.2, "Adding a Global Attribute"
- Section 55.3, "Editing a Global Attribute"
- Section 55.4, "Deleting a Global Attribute"

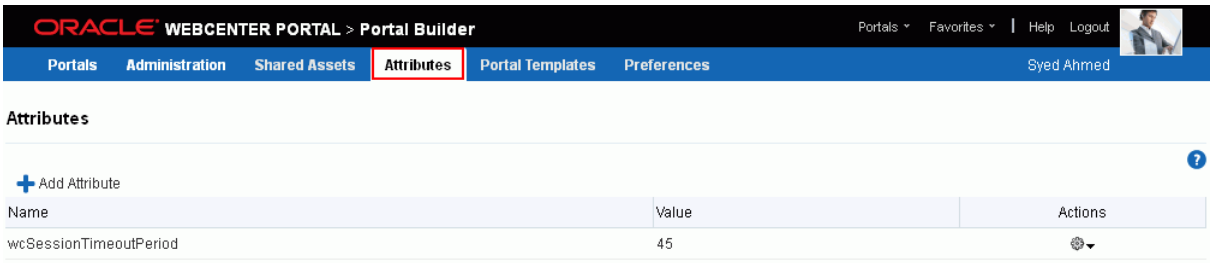
> **Permissions:** To perform the tasks in this chapter, you must have the WebCenter Portal `Administrator` role or a custom role that grants at least the following permission:
>
> - `Portal Server-Manage Configuration`
>
> For more information about permissions, see Section 49.3, "About Application Roles and Permissions."

55.1 About Global Attributes

Every portal includes built-in attributes such as name, description, date created, icon, and so on. In addition to these built-in attributes, portal moderators can add custom attributes that are unique to the portal and its characteristics to specify additional portal information (metadata). Custom attributes are propagated throughout a portal. For information about working with attributes unique to a specific portal, see the "Administering Attributes in a Portal" chapter in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

In addition to portal-specific attributes, system administrators can add and manage global attributes from the **Attributes** page (Figure 55–1) in Portal Builder administration. Global attributes are available for use by any portal.

Figure 55–1 Portal Builder Administration: Attributes



A custom attribute is simply a name value pair (such as `customerId=400`, `orderId=11`, or `userName=Smith`). A custom attribute name is unique within a particular portal. For example, you can use a global attribute in a portal for customer analysis purposes with several custom task flows that take the parameter `customerId` as an input: task flows such as Customer Sales History, Customer Satisfaction Rating, Future Sales Prospects, or Customer Contact Information. With a custom attribute defined named `customerId` with an appropriate value, all the task flows that can accept a `customerId` can display information specific to that customer.

A custom attribute can also be retrieved using Expression Language (EL) expressions. For example, an EL expression may read a value that is passed in through the URL that displays a portal (for example, `customerid=10`). Any portal pages, task flows, or portlets that deliver customized content based on parameter values can accept global custom attribute values and display content accordingly using the following Expression Language (EL) syntax to access the global custom attribute value:

```
#{WCAppContext.application.applicationConfig.customAttributes[attributeName]
```

If you need EL assistance, an application developer can provide an EL expression; see the "Expression Language Expressions" appendix in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

55.2 Adding a Global Attribute

To add a new global attribute for use by any portal:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Attributes**.

   You can also enter the following URL in your browser to navigate directly to the **Attributes** page:

   ```
   http://host:port/webcenter/portal/builder/attributes
   ```

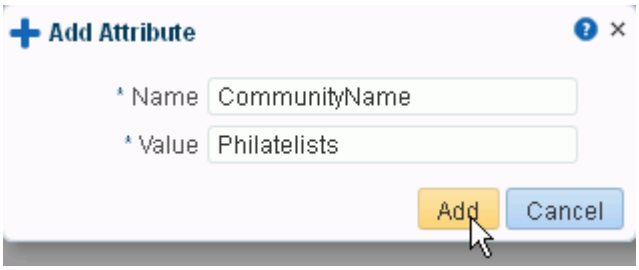
   See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click **Add Attribute** (Figure 55–2).

Figure 55–2 WebCenter Portal Administration - Add Attribute

The Add Attribute dialog opens (Figure 55–3).

Figure 55–3 Entering Custom Attribute Name and Value



3. Enter a unique **Name** for the attribute. Valid names start with an alphabetic character and contain only alphanumeric characters
4. Optionally, enter a **Value** for the custom attribute. The value you type is treated as a string value.
5. Click **Add** to save the custom attribute and display it in the list on the **Attributes** page.

55.3 Editing a Global Attribute

To edit a global attribute:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Attributes**.

   You can also enter the following URL in your browser to navigate directly to the **Attributes** page:

   `http://host:port/webcenter/portal/builder/attributes`

   See Also: "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click the **Actions** icon for the attribute and select **Edit Attribute**.
3. In the Edit Attribute dialog, modify the attribute **Value**. The value you type is treated as a string value.
4. Click **OK** to save your changes.

55.4 Deleting a Global Attribute

To delete a global attribute:

1. On the **Administration** page (see Section 47.2, "Accessing the Portal Builder Administration Page"), click **Attributes**.

   You can also enter the following URL in your browser to navigate directly to the **Attributes** page:

   `http://host:port/webcenter/portal/builder/attributes`

   > **See Also:** "WebCenter Portal Pretty URLs" appendix in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

2. Click the **Actions** icon for the attribute and select **Delete Attribute**.
3. In the confirmation dialog, click **Delete**.

56

Analyzing Portal Usage

The Analytics service offers real-time usage and activity reporting for your portal. This chapter describes how to use the pages and task flows provided through the Analytics service. It includes the following sections:

- Section 56.1, "About the Analytics Task Flows and Service"
- Section 56.2, "About the Analytics Administration Page"
- Section 56.3, "Working with Analytics Task Flows"

For Analytics task flows to work, the Analytics schema (ACTIVITIES) must be installed and configured, and a connection set up between your application and the Analytics Collector. For more information about the Analytics schema and how to manage the Analytics service backend, see Chapter 11, "Managing Analytics."

This chapter describes how to add Analytics task flows to application pages at runtime; to learn how to add Analytics task flows at design time using Oracle JDeveloper, see the "Integrating Analytics" chapter in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

Permissions: To perform the tasks in this chapter, you must have the WebCenter Portal `Administrator` role or a custom role that grants the following permission:

- `Portal Server-Manage Configuration`

Additionally, you need permissions to create and manage portals (`Portals-Create` and/or `Portals-Manage All`). For more information about permissions, see Section 49.3, "About Application Roles and Permissions."

56.1 About the Analytics Task Flows and Service

The Analytics service allows WebCenter Portal administrators and moderators to track and analyze WebCenter Portal traffic and usage. The Analytics service provides the following basic functionality:

- Usage Tracking Metrics: The Analytics service collects and reports metrics of common WebCenter Portal functions, including community and portlet traffic.
- Behavior Tracking: The Analytics service can be used to analyze WebCenter Portal metrics to determine usage patterns, such as page visit duration and usage over time.

- User Profile Correlation: The Analytics service can be used to correlate metric information with user profile information. Usage tracking reports can be viewed and filtered by user profile data such as country, company or title.

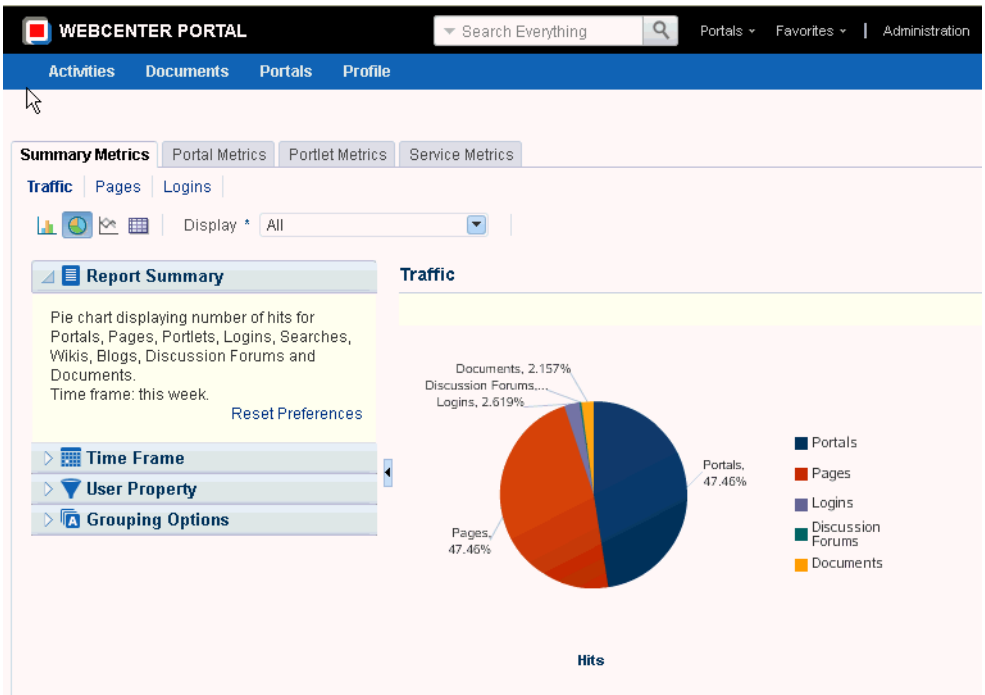
> **Note:** Profile information is cached meaning that changes to a user profile are not visible in reports until the cache is updated. The default cache time is 60 minutes, but this value can be changed by your administrator.

56.2 About the Analytics Administration Page

An *analytics console* that displays metrics for the entire WebCenter Portal is available to WebCenter Portal administrators with the `Manage Configuration` permission. The console consists of four pages, grouping several different reports:

- **Summary Metrics** - portal traffic, page views, and login metrics
- **Portal Metrics** - Portal usage and response times
- **Portlet Metrics** - Portlet views and response times
- **Service Metrics** - Usage of searches, documents, wikis, blogs and discussions

Figure 56–1 Analytics Console for Administrators



Out-of-the-box, this console is only available through a business role page named *Analytics*. It is the WebCenter Portal administrator's responsibility to grant people permissions to see the Analytics page. This page is intended for anyone who needs to analyze access and usage statistics; this could include administrators, sales or marketing managers or directors, business analysts, and so on.

Just like other business role pages, the Analytics page is pushed to all the users to whom it is assigned, appearing in the Home portal. Once the Analytics page is

available in the Home portal, users can show and hide the page through the Manage Page dialog.

56.3 Working with Analytics Task Flows

This section describes the Analytics task flows, including how to add them to a portal page, how to customize them, how to change their properties, and how to personalize report views.

This section contains the following topics:

56.3.1 Understanding Analytics Task Flows

This section lists and describes all the Analytics task flows that are provided with WebCenter Portal. Note that those marked with "Administrator" are only available to users with an Administrator account, and those marked with "System Administrator" are only available to system administrators.

The following task flows are available out-of-the-box:

Application Analytics:

Portal Analytics:

Portlet Analytics:

Service Analytics:

Note: The images shown in the following sections represent one view of each report. However each report can be customized to display the data in different ways (for example, a bar chart, a pie chart, a line chart, or a table). For information on customizing reports, see Section 56.3.3, "Customizing Analytics Reports"and Section 56.3.4, "Personalizing Your Analytics Report."

56.3.1.1 WebCenter Traffic

The WebCenter Traffic task flow displays a summarized view for common events within the portal. Use this task flow to track application-wide events—portal views, page views, portlet views, logins, number of searches, wiki views, blog views, discussion forum views, and document views. For more information, see the "WebCenter Traffic" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

56.3.1.2 Page Traffic (Administrator)

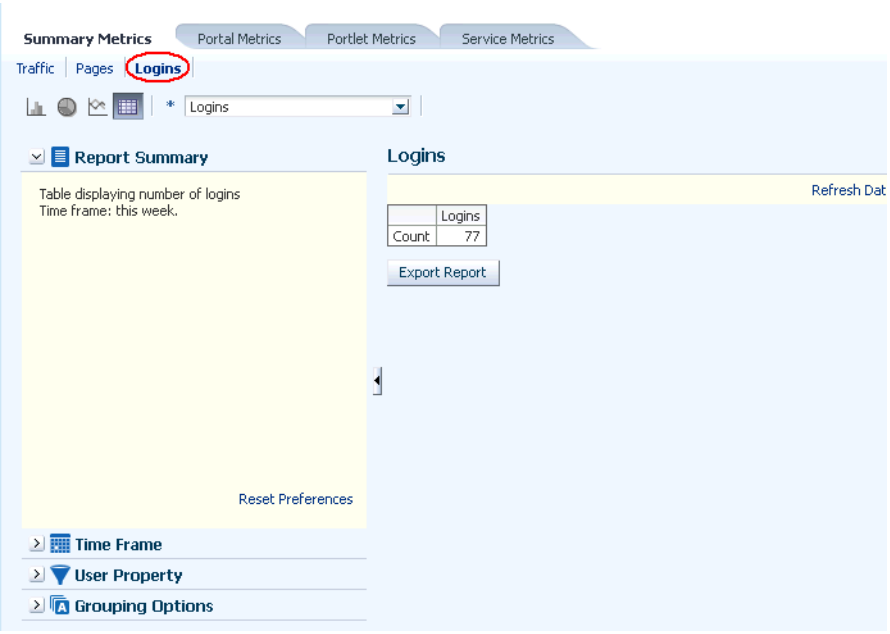
The Page Traffic task flow displays the number of page hits and the number of unique users that have visited any portal page. Use this task flow to quickly see the most visited pages (top pages) and/or the least visited pages (bottom pages). For more information, see the "Page Traffic (Administrator)" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

56.3.1.3 Login Metrics (System Administrator)

The Login task flow (Figure 56–2) reports the number of times users log in to WebCenter Portal.

Use this task flow to see the total number of portal logins and/or the number of times unique users logged into WebCenter Portal.

Figure 56–2 Analytics Task Flow - Login Metrics

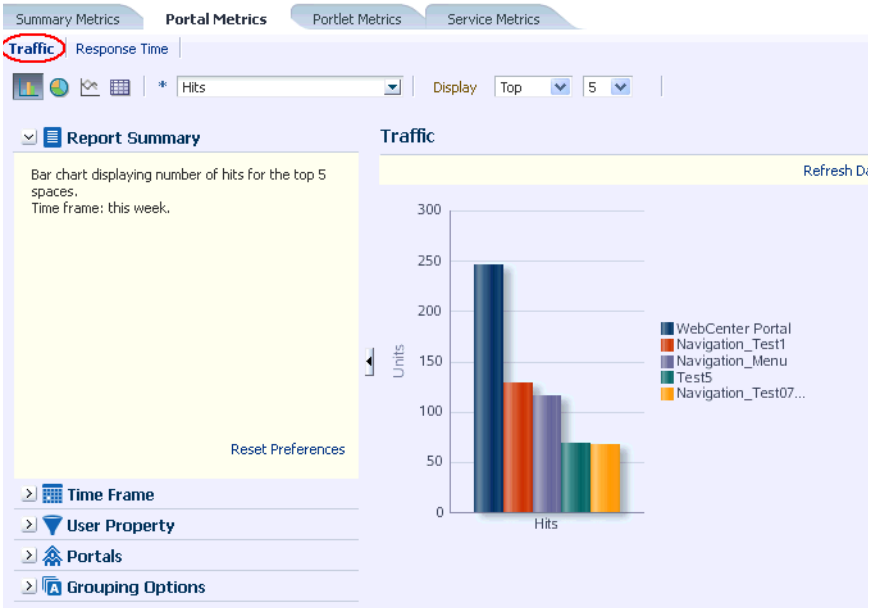


56.3.1.4 Portal Traffic (System Administrator)

The Portal Traffic task flow (Figure 56–3) displays usage information—the number of page hits, number of unique users, and the number of unique visits (multiple consecutive page views within the same portal during the same WebCenter Portal session is treated as one visit)—for individual portals.

Use this task flow to quickly see the most popular portals (top), and the least popular portals (bottom). You can filter the data to only show specific portals or show all portals.

Figure 56–3 Analytics Task Flow - Portal Traffic

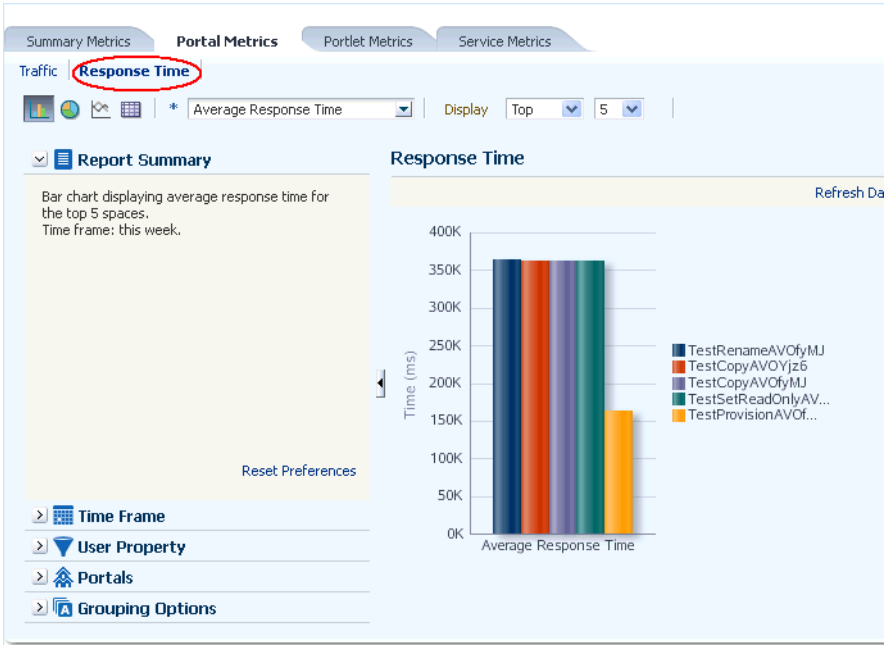


Note: The Home portal is not included in the data.

56.3.1.5 Portal Response Time (System Administrator)

The Portal Response Time task flow (Figure 56–4) displays page performance information—average, minimum, or maximum response time—for individual portals over any time period you specify. Use this task flow to quickly see the slowest portals (bottom), and the fastest portals (top). You can filter the data to only show specific portals or show all portals.

Figure 56–4 Analytics Task Flow - Portal Response Time



Note: The Home Portal is not included in the data.

56.3.1.6 Portlet Traffic (Administrator)

The Portlet Traffic task flow displays portlet usage information—the number of portlet hits (the number of times a portlet is displayed) and number of unique users that access a portlet.

Use this task flow to quickly see the most popular portlets (top), and the least popular portlets (bottom). You can filter the data to only show specific portlets or show all portlets. Similarly, you can filter the portlet data by portal. For more information, see the "Portlet Traffic (Administrator)" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

56.3.1.7 Portlet Instance Traffic (Administrator)

The Portlet Instance Traffic task flow displays usage information—the number of portlet hits (the number of times a portlet is displayed) and number of unique users that access a portlet—for individual portlet instances. If the same portlet displays on several different pages, each placement is considered as a portlet instance.

Use this task flow to quickly see the most popular portlet instances (top), and the least popular portlet instances (bottom). You can filter the data to only show specific portlet instances or show all portlet instances. Similarly, you can filter the portlet data by portal. For more information, see the "Portlet Instance Traffic (Administrator)" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

56.3.1.8 Portlet Response Time (Administrator)

The Portlet Response Time task flow displays performance information—average, minimum, and maximum response time—for individual portlets. Use this task flow to quickly see the slowest portlets (bottom), the fastest portlets (top), and compare performance data. Portlet response times are important because there is often a direct

link between page performance and the slowest portlets. When troubleshooting poor performance within a portal, it is important to identify the worst performing portlets. You can filter the data to only show specific portlets or show all portlets. Similarly, you can filter the portlet data by portal. For more information, see the "Portlet Response Time (Administrator)" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

56.3.1.9 Portlet Instances Response Time (Administrator)

The Portlet Instances Response Time task flow displays performance information—average, minimum, and maximum response time—for individual portlet instances. If the same portlet displays on several different pages, each placement is considered as a portlet instance.

Use this task flow to quickly see the slowest portlet instances (bottom), the fastest portlet instances (top), and compare performance data. You can filter the data to only show specific portlet instances or show all portlet instances. Similarly, you can filter the portlet data by portal. For more information, see the "Portlet Instances Response Time (Administrator)" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

56.3.1.10 Search Metrics

The Search Metrics task flow tracks searches performed within the portal. Use this task flow to quickly see the most popular (top) and least popular (bottom) search phrases. For more information, see the "Search Metrics"section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

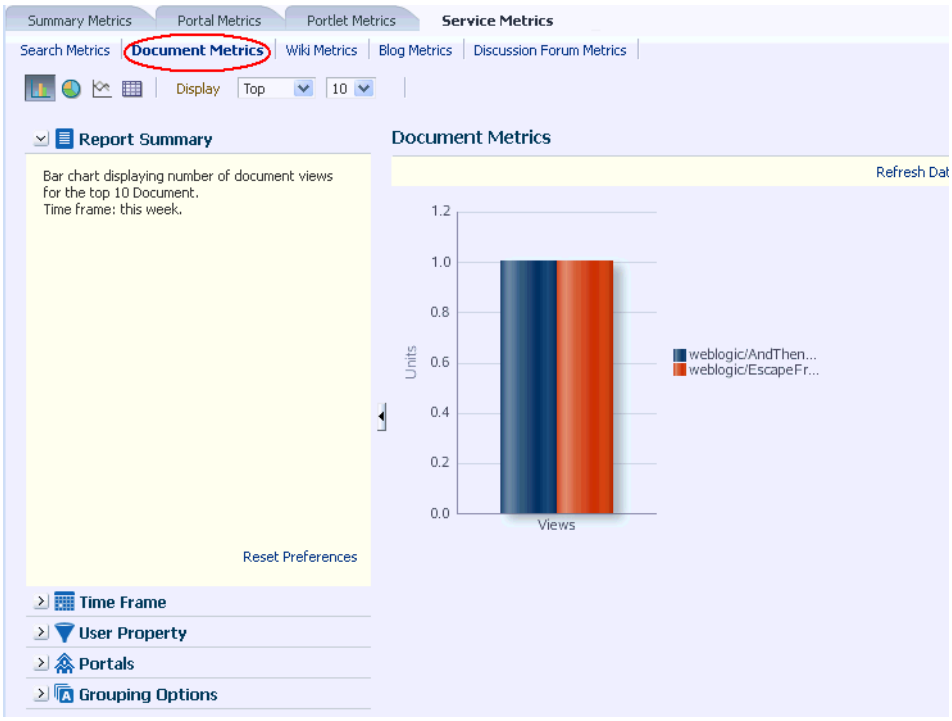
56.3.1.11 Document Metrics (System Administrator)

The Document Metrics task flow (Figure 56–5) tracks how often a document is accessed. Use this task flow to quickly see the most popular (top) and least popular (bottom) documents. You can filter the data to only show specific portals or show all portals.

Note: Documents in the Home Portal are included in this report.

Note: If you have two different documents with the same name, they are treated as two separate documents. The metrics include the parent folder for context.

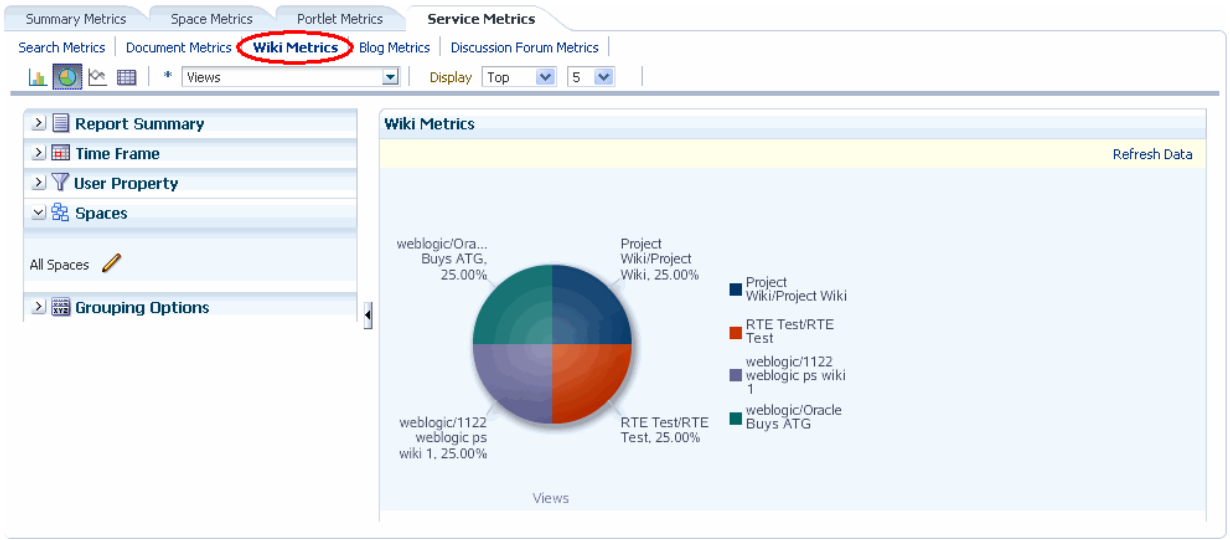
Figure 56–5 Analytics Task Flow - Document Metrics



56.3.1.12 Wiki Metrics (System Administrator)

The Wiki Metrics task flow (Figure 56–6) tracks how often wikis are accessed within WebCenter Portal. Use this task flow to quickly see the most popular (top) and least popular (bottom) wikis. You can filter the data to only show specific portals or show all portals.

Figure 56–6 Analytics Task Flow - Wiki Metrics

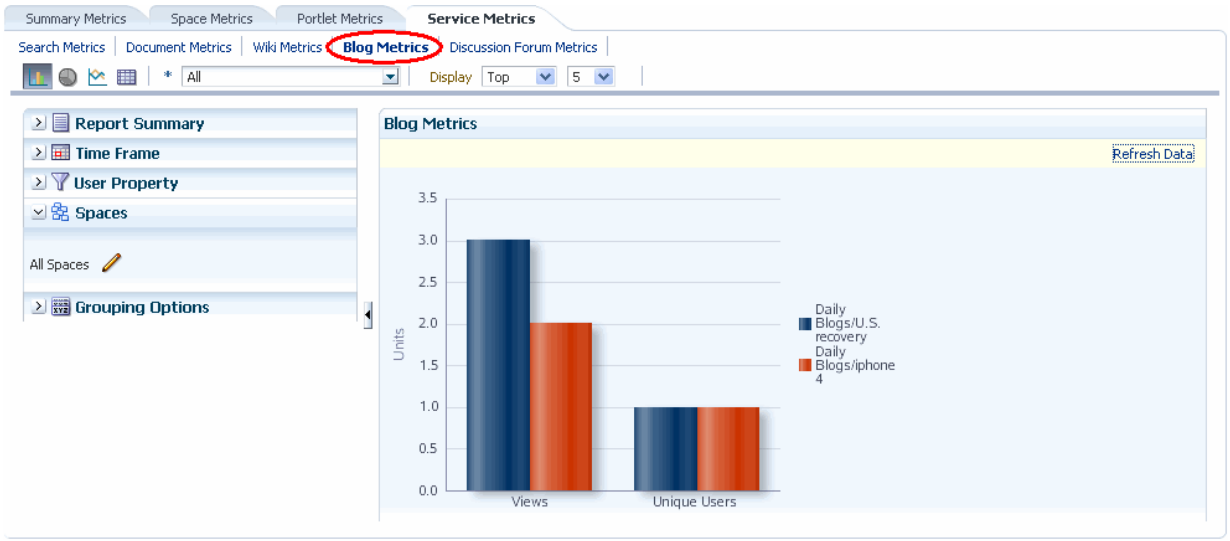


56.3.1.13 Blog Metrics (System Administrator)

The Blog Metrics task flow (Figure 56–7) tracks how often blogs are accessed within WebCenter Portal. Use this task flow to quickly see the most popular (top) and least

popular (bottom) blogs. You can filter the data to only show specific portals or show all portals.

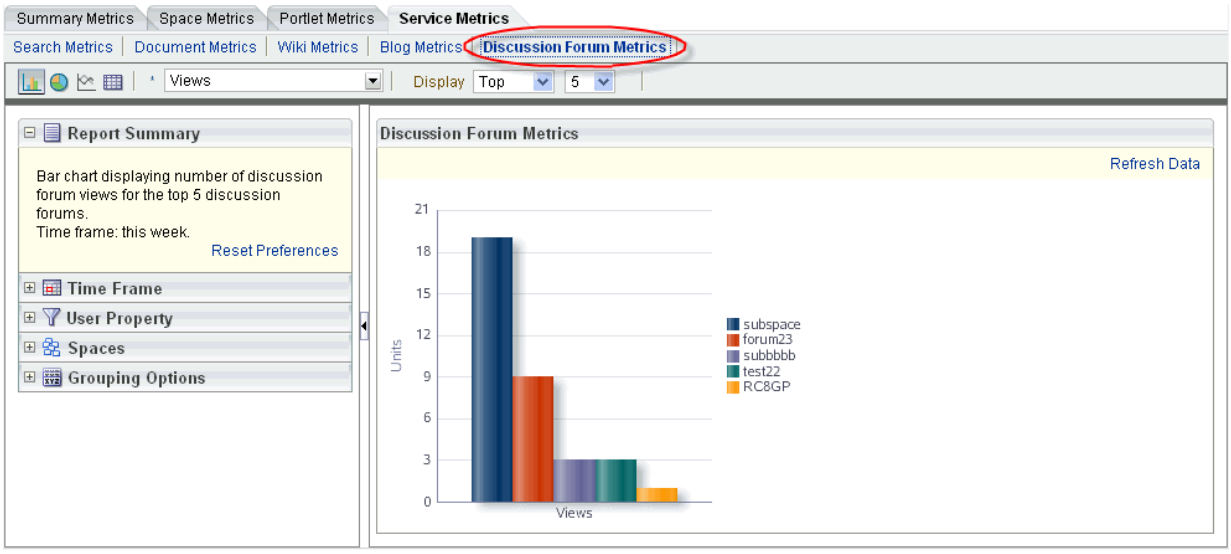
Figure 56–7 Analytics Task Flow - Blog Metrics



56.3.1.14 Discussion Forum Metrics (System Administrator)

The Discussion Forum Metrics task flow (Figure 56–8) tracks discussion forums within WebCenter Portal. Use this task flow to quickly see the most popular (top) and least popular (bottom) discussions. You can filter the data to only show specific portals or show all portals.

Figure 56–8 Analytics Task Flow - Discussion Metrics

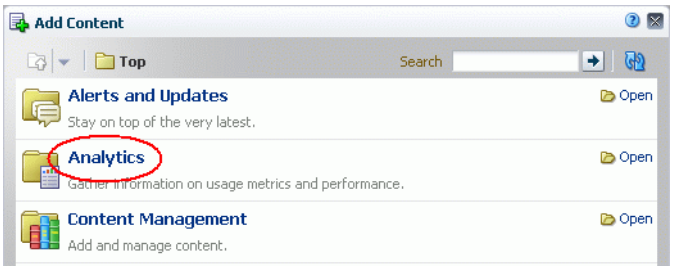


56.3.2 Adding Analytics Task Flows to a Page

The process of adding an Analytics task flow to a page is the same as for any other task flow (for more information, see the "Adding a Component to a Page" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*). The process

varies only in where you can find these task flows in the Resource Catalog. All the Analytics task flows are under the **Analytics** folder.

Figure 56–9 Analytics Folder in Resource Catalog



Note: When you add an Analytics task flow to a portal, it displays information only for that portal, not for all portals.

56.3.3 Customizing Analytics Reports

You can set defaults for Analytics reports by editing the report settings in page Edit mode. Any changes you make while in Edit mode will become the default report settings for all users in page View mode.

For example, you can edit the Analytics page, changing the following settings on the **Summary Metrics** page in the Traffic report: set the report type to pie chart, set the time frame to this week, and remove Discussion Forums from the display. When users visit the Analytics page, those settings will be applied by default. Users can then edit the report as necessary for their needs. This can be useful if there are particular settings you know are commonly used by your users, or to customize a particular instance of an Analytics task flow on a group-specific page.

You can also configure report settings to specify the controls available to users in View mode. For more information, see the "Customizing Analytics Reports" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

56.3.4 Personalizing Your Analytics Report

Analytics task flows include display options at the top of the report and query options to the left of the report. These options enable you to personalize the report for your needs by changing the metrics included in the report and the way the report is presented. Most options are the same for all Analytics task flows.

This section includes the following subsections:

- Section 56.3.4.1, "Report Display Options"
- Section 56.3.4.2, "Query Options"

56.3.4.1 Report Display Options

The report display options at the top of the report enable you to select the type of report, select the type of metrics to include, and, for some task flows, control the top/bottom range to display.

Report Types

You can display your report as a bar chart, pie chart, line chart, or table depending on the display and query options you select. To choose your report type, click the associated icon.

Table 56–1 lists the report types available for different display and query options. It includes the following columns:

- Selected Metrics specifies what has been selected in the list of metrics, a single metric or multiple metrics.

> **Note:** Search Metrics and Document Metrics task flows show only those single metrics; there is no list to select metrics.

- Group By Options specifies what has been selected in the Grouping Options section to the left of the report, **No Selection** or one of the available selections.
- Bar, Pie, Line, and Table specify whether you can view that type of report with the specified selections.

Table 56–1 Display Options for the Analytics Task Flows

Selected Metrics	Group By Option	Bar	Pie	Line	Table
Single metric Login Traffic task flow	No selection	N	N	N	Y
Single metric All other task flows	No selection	Y	Y	N	Y
Single metric	Time interval, user property, or Both*	Y	N	Y	Y
Multiple metrics WebCenter Traffic and Login Traffic task flows	No selection	Y	Y	N	Y
Multiple metrics All other task flows	No selection	Y	N	Y	Y
Multiple metrics WebCenter Traffic and Login Traffic task flows	Time interval or user property	Y	N	Y	Y
Multiple metrics All other task flows	Time interval or user property	N	N	N	Y
Multiple metrics Login Traffic task flow	Both*	N	N	N	Y

* The grouping option **Both** is available only for the Login Traffic task flow.

Metrics

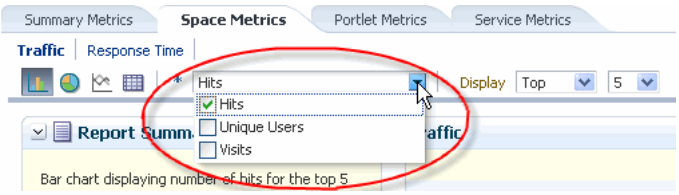
You can select which type of metrics to include in your report. Your metrics options differ depending on the task flow you are using:

- WebCenter Traffic: Spaces, Pages, Portlets, Logins, Searches, Wikis, Blogs, Discussion Forums, Documents
- Page Traffic: Hits, Unique Users
- Login Metrics: Logins, Unique Users
- Space Traffic: Hits, Unique Users, Visits

- Space Response Time: Average Response Time, Minimum Response Time, Maximum Response Time
- Portlet Traffic: Hits, Unique Users
- Portlet Instance Traffic: Hits, Unique Users
- Portlet Response Time: Average Response Time, Minimum Response Time, Maximum Response Time
- Portlet Instance Response Time: Average Response Time, Minimum Response Time, Maximum Response Time
- Search Metrics: This task flow shows only search metrics, so it does not include an option to select metrics.
- Document Metrics: This task flow shows only document metrics, so it does not include an option to select metrics.
- Wiki Metrics: Views, Unique Users
- Blog Metrics: Views, Unique Users
- Discussion Forum Metrics: Views, Unique Users

To select which metrics to include in your report, select the metrics from the list above the report.

Figure 56–10 Analytics Task Flow - Metrics Selection



Top, Bottom, or Custom Ranges

With some task flows you can specify whether you want to see the top, bottom, all, or a custom ranges of metrics in your report. Use these options to see the most and least popular items in your portal.

To display the top or bottom ranges of metrics in your report, in the lists above the report, select **Top** or **Bottom**, and then select a number to define the range.

To display a custom range, in the list above the report, select **Specify**, then click **Select**.

The top and bottom options are available for Pages, Portlet Traffic, Portlet Instances Traffic, Response Time, Portlet Response Time, Portlet Instances Response Time.

The custom range option is available for Pages, Traffic, Response Time, Portlet Traffic, Portlet Instances Traffic, Response Time, Portlet Response Time, Portlet Instances Response Time, Search Metrics, Document Metrics, Wiki Metrics, Blog Metrics, Discussion Forum Metrics.

56.3.4.2 Query Options

Analytics task flows include the following query options to the left of the report:

- **Report Summary**

  Displays a summary of the selected display and query options shown in the report.

- **Time Frame**

  Enables you to specify the date range for the metrics displayed in the report. You can select from the following options: Yesterday, Today, This Week, Last Week, This Month, Last Month, Last Three Months, Last Six Months, This Year, Last Year, or you can specify your own date range.

- **User Property**

  Enables you to filter your report by user property. After selecting a property from the list, you can specify a value that the property must contain or must not contain, and only metrics that apply to the filtered property display in the report.

  – **Property**: Select a property on which to filter the report. You can select City, Company, Country, Department, Display Name, Employee ID, IM User, Manager, Phone, State or Province, Street, Title, or ZIP code

  – **Operator**: Select how you want to filter the property. You can select **Contains** or **Does Not Contain**.

  – **Value**: Type a value on which to filter the property.

  Note: To search using a wildcard (for example, % or ?), you must prefix the wildcard with a forward slash (\). For example, to search for give or giving, type `giv\%` in the **Value** box.

- **Additional Options**

  Enables you to include Home portal pages in report data. These options are available with the Pages task flow (in the Page Traffic report).

- **Spaces**

  When Analytics task flows display in the Home portal or on a business role page, you can choose which spaces to include in your report. When Analytics task flows are used within a particular space, only metrics only for that space display; the Spaces option is unavailable (grayed out).

  To specify the spaces to include in your report, click the **Space Filter** icon to display the Specify Spaces popup. Select the spaces you want to include in your report, using CTRL+click and SHIFT+click to select multiple spaces.

  This option is not available with the Traffic, Logins, or Search Metrics task flows.

- **Grouping Options**

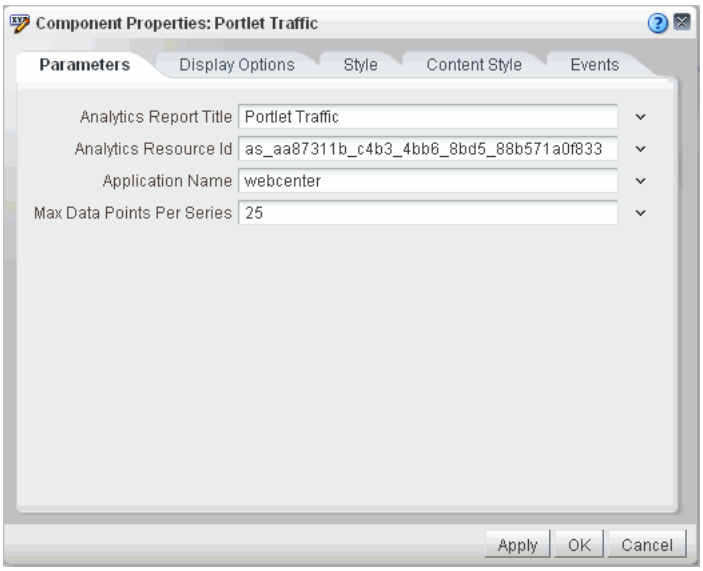
  Enables you to select an option by which to group the metrics in your report. You can group by a time interval (Hour, Day, Week, Month, or Year), a user property, or, with the Logins task flow, both.

  Note: This setting affects the available display options for the report (see Table 56–10).

56.3.5 Setting Analytics Task Flow Properties

The Analytics service task flows have associated properties, which users with sufficient privileges can access through the Component Properties dialog in Composer (Figure 56–11).

Figure 56–11 Analytics Task Flow - Component Properties



For information about accessing the Component Properties dialog, see the "Setting Properties on a Component" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

The following sections provide information about properties of the Events service task flows and describe the properties on the Parameters tab:

- Section 56.3.5.1, "About the Analytics Service Task Flow Properties"
- Section 56.3.5.2, "Analytics Service Task Flow Parameters"

56.3.5.1 About the Analytics Service Task Flow Properties

The properties on the **Parameters** tab of the Component Properties dialog control the default task flow content. For descriptions of the parameters on this tab, see Section 56.3.5.2, "Analytics Service Task Flow Parameters." For some task flows, parameters on this tab facilitate the wiring of the task flow to page parameters and page definition variables. For information about wiring pages and components, see the "Wiring Pages, Task Flows, Portlets, and UI Components" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

Changes to the properties on the **Display Options**, **Style**, and **Content Style** tabs affect the appearance and behavior of the task flow. These properties are common to all task flows. For more information, see the "Modifying Components" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

The contents of the **Events** tab depend on the events supported by the task flow. For more information, see the "Working with Component Contextual Events" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*.

All properties on the **Parameters** and **Display Options** tabs provide access to an Expression Language (EL) editor, which you can use to select or specify a variable value instead of a constant value. Click the **Edit** icon next to a property field to open the editor.

Note: Wherever you enter EL expressions on the generic Display Options tab in the Component Properties dialog, the entry is automatically validated. If the EL syntax is invalid, an error appears and the value is neither applied nor saved. Generic Display Options are those cataloged in the "Display Options Properties" section in *Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal*. For more information about ELs in WebCenter Portal, see the "Expression Language Expressions" appendix in *Oracle Fusion Middleware Developing Portals with Oracle WebCenter Portal and Oracle JDeveloper*.

EL validation is not performed on non-generic display options.

56.3.5.2 Analytics Service Task Flow Parameters

Table 56–2 describes the parameters that are unique to the Analytics service task flows.

Table 56–2 Analytics Task Flow Parameters

Parameter	Description
Analytics Report Title	Specifies the display title that appears above the analytics data. **Note:** ■ Use the Analytics Report Title rather than the Text property on the **Display Options** page. Changing the Text value has no effect on Analytics task flows. ■ You cannot change the report titles in the Analytics console.
Analytics Resource Id	Specifies the MDS document used to store user customizations/application customizations for the task flow instance in MDS. **Warning:** Do not edit this value.
Application Name*	Specifies the WebCenter Portal application for which you want to display analytics data. For WebCenter Portal, this is always `webcenter`. The analytics database can be used to store event data from multiple applications so this parameter is required to identify which application data to display. If omitted, the task flow displays analytics data for all supported WebCenter Portal applications.
Max Data Points Per Series	Indicates the maximum number of data points to be displayed in a bar or line chart. The default value is 25. Valid values are between 1 and 1000. **Note:** Increasing the number of data points might increase the time it takes to render the report.

Part XII

Appendixes

This part presents appendix information for *Oracle Fusion Middleware Administering Oracle WebCenter Portal*.

Part XII contains the following appendixes:

- Appendix A, "Oracle WebCenter Portal Configuration"
- Appendix B, "Oracle HTTP Server Configuration for WebCenter Portal"
- Appendix C, "Third-Party Product Support"
- Appendix D, "Oracle Secure Enterprise Search Configuration for Evaluation"
- Appendix E, "Labeling During WebCenter Portal Lifecycle"
- Appendix F, "Migrating Wiki Content to WebCenter Portal"
- Appendix G, "Troubleshooting Oracle WebCenter Portal"

A

Oracle WebCenter Portal Configuration

This chapter describes the two main configuration files for WebCenter Portal and Portal Framework applications `adf-config.xml` and `connections.xml`. This appendix describes both these files, how to locate them, and also when to configure these files and which tools to use.

Other configuration files, such as `web.xml` and `webcenter-config.xml` are described here too. See also Section 1.3.5, "Oracle WebCenter Portal Configuration Considerations."

This appendix includes the following topics:

- Section A.1, "Configuration Files"
- Section A.2, "Cluster Configuration"
- Section A.3, "Configuration Tools"

See Appendix G.2, "Troubleshooting Oracle WebCenter Portal Configuration Issues."

A.1 Configuration Files

`adf-config.xml`, `connections.xml`, and `web.xml` are used to configure WebCenter Portal and Portal Framework applications and their back-end services. In addition, the `webcenter-config.xml` configuration file, which is specific to the out-of-the-box application WebCenter Portal, is used to configure application-wide settings.

This section describes how applications use each file and the location of these files post deployment. This section includes the following subsections:

- adf-config.xml and connections.xml
- web.xml
- webcenter-config.xml

A.1.1 adf-config.xml and connections.xml

`adf-config.xml` and `connections.xml` both store design time configuration information, such as the discussions server, mail server, or content server that is used by the application in the development environment:

- **adf-config.xml** - Stores application-level settings, such as which discussions server or mail server the application is currently using.

  See *Oracle Fusion Middleware Fusion Developer's Guide for Oracle Application Development Framework*.

- **connections.xml** - Stores connection details for WebCenter Portal services.

  See *Oracle Fusion Middleware Fusion Developer's Guide for Oracle Application Development Framework*.

After you deploy WebCenter Portal or a Portal Framework application to a production environment, Oracle recommends that you use Fusion Middleware Control or WebLogic Scripting Tool (WLST) commands to reconfigure properties in these files. For example, you may want to modify connection details to point to production server instances. See Appendix A.3, "Configuration Tools."

The main advantage of using Fusion Middleware Control and WLST commands is that any configuration changes that you make, post deployment, are stored as *customizations* in the application's Oracle Metadata Services (MDS) repository. MDS uses the original deployed versions of `adf-config.xml` and `connections.xml` as base documents and stores all subsequent customizations separately into MDS using a single customization layer. If the application is redeployed in the future, all previous configuration changes are retained.

When WebCenter Portal or a Portal Framework application starts up, application customizations stored in MDS are applied to the appropriate base documents and the application uses the merged documents (base documents with customizations) as the final set of configuration properties.

This section includes the following subsections:

- Reviewing Post Deployment Customizations in MDS
- Exporting Configuration Files with MDS Customizations
- Handling Configuration Conflicts
- Deleting MDS Customizations for adf-config.xml or connections.xml
- What Configuration Tool to Use

For more information on MDS customizations, see "Understanding the MDS Repository" in *Oracle Fusion Middleware Administrator's Guide*.

Reviewing Post Deployment Customizations in MDS

Post deployment, always use Fusion Middleware Control or WLST commands to review the latest configuration or make configuration changes. In Fusion Middleware Control you will mostly use WebCenter Portal-specific configuration screens but a useful Systems MBean Browser is also available for reviewing configuration settings. These tools always show you the current configuration so, typically, there is no need for you to examine or change the content of base documents or MDS customization data for files such as `adf-config.xml` and `connections.xml`.

At times it might be useful to 'see' the information in MDS. If for any reason you must extract or examine configuration file customizations that are stored in MDS, use the WLST command `exportMetadata`.

> **See Also:** For detailed syntax and examples, see "exportMetadata" in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For example, to determine MDS customizations for `connections.xml` in WebCenter Portal, which has the application name `webcenter` and is deployed to the `WC_Spaces` managed server, the file name and location is always `/META-INF/mdssys/cust/adfshare/adfshare/connections.xml.xml`, you might specify:

```
exportMetadata(application='webcenter', server='WC_Spaces',
 toLocation='/tmp/mydata',
 docs='/META-INF/mdssys/cust/adfshare/adfshare/connections.xml.xml')
```

And similarly, to determine MDS customizations for `adf-config.xml`:

```
exportMetadata(application='webcenter', server='WC_Spaces',
 toLocation='/tmp/mydata',
 docs='/META-INF/mdssys/cust/adfshare/adfshare/adf-config.xml.xml')
```

You choose where to save file customizations by specifying `toLocation`. If, for example, `toLocation` is set to `/tmp/mydata`, then the requested file is saved to `/tmp/mydata/META-INF/mdssys/cust/adfshare/adfshare`.

If no customizations exist for the requested file, then nothing is saved to the specified location—previously extracted customizations at the same location are not overwritten.

Exporting Configuration Files with MDS Customizations

You can use the System MBean Browser to obtain "current versions" of configuration files such as `adf-config.xml` or `connections.xml`, that is, a version of the file that includes the base document merged with MDS customizations.

To export `adf-config.xml` or `connections.xml` with MDS customizations from the System MBean Browser:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal or your Portal Framework application. For more information, see:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For the WebCenter Portal application - From the **WebCenter Portal** menu, select **System MBean Browser**.
   - For Portal Framework applications - From the **Application Deployment** menu, select **System MBean Browser**.
3. Expand **Application Defined MBeans**.
4. Navigate to the MBean associated with the file you want to export.

   For example, navigate to MBeans for `adf-config.xml` or `connections.xml` as follows:

   - `adf-config.xml` - Click **oracle.adf.share.config** > **Server: name**> **Application: name**> **ADFConfig**> **ADFConfig**

     For the WebCenter Portal application: **oracle.adf.share.config** > **Server: WC_Spaces**> **Application: webcenter**> **ADFConfig**> **ADFConfig**
   - `connections.xml` - Click **oracle.adf.share.connections**> **Server: name**> **Application: name**> **ADFConnnections**> **ADFConnections**

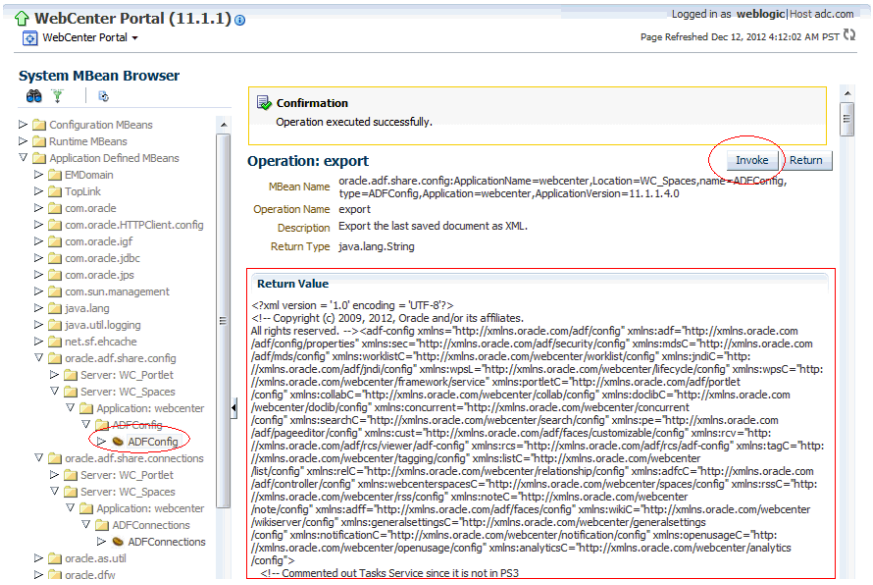
     For the WebCenter Portal application: **oracle.adf.share.config** > **Server: WC_Spaces**> **Application: webcenter**> **ADFConnections**> **ADFConnections**
5. Click the **Operations** tab.
6. Click **Export**.

Alternatively, click **ExportToDisk** and then specify a sever location for the XML file.

7. Click **Invoke**.

   If you selected the **Export** operation, the content of the XML file displays on the screen (Figure A–1).

Figure A–1 Exporting Configuration Files with MDS Customizations



Handling Configuration Conflicts

MDS customizations use references to elements in the base document to call out which elements must be inserted/deleted/replaced, and at what location. If an element is inadvertently removed from a future redeployment and MDS contains a reference to that element, then the WebCenter Portal or Portal Framework application's configuration appears corrupt.

For example, consider a Portal Framework application built using JDeveloper called `MyPortalApp`, with a connection, created at design-time, called `myconnection`. The application was deployed to a managed server, and a URL in `myconnection` was modified. This modification is stored in MDS as a customization instruction to update `myconnection` to use the new URL. If in the future, `myconnection` is removed at design time and the application redeployed using the same MDS details, a configuration conflict occurs, that is, the customization instruction in MDS attempts to find `myconnection` but no such configuration exists.

You are unlikely to face this problem but should a previously deployed application appear corrupt after making changes to `adf-config.xml` or `connections.xml` you have the following options:

- Remove the MDS customization causing conflict manually:

  1. Extract MDS customization information for `adf-config.xml` or `connections.xml`.

     For example, for WebCenter Portal specify:

     ```
     exportMetadata(application='webcenter', server='WC_Spaces',
      toLocation='/tmp/mydata',
      docs='/META-INF/mdssys/cust/adfshare/adfshare/adf-config.xml.xml')
     ```

```
exportMetadata(application='webcenter', server='WC_Spaces',
 toLocation='/tmp/mydata',
 docs='/META-INF/mdssys/cust/adfshare/adfshare/connections.xml.xml')
```

2. Remove the customization instruction that is causing conflict from the document.
3. Import the modified document back in to MDS.

   For example, for WebCenter Portal specify:

```
importMetadata(application='webcenter', server='WC_Spaces',
 fromLocation='/tmp/mydata',
 docs='/META-INF/mdssys/cust/adfshare/adfshare/connections.xml.xml')

importMetadata(application='webcenter', server='WC_Spaces',
 fromLocation='/tmp/mydata',
 docs='/META-INF/mdssys/cust/adfshare/adfshare/adf-config.xml.xml')
```

4. Restart the managed server.

- Delete MDS customizations for `adf-config.xml` or `connections.xml`, deploy the new EAR file, and reconfigure your application from scratch using Fusion Middleware Control or WLST.

  See below for detailed steps, "Deleting MDS Customizations for adf-config.xml or connections.xml".

- Redeploy the EAR file on a new partition or a partition where older customizations are deleted. In either case, all data previously stored in MDS for the application is lost, including any application customizations for `adf-config.xml` or `connections.xml`, and all user customizations. You must reconfigure your application from scratch too, using Fusion Middleware Control or WLST.

  See "deleteMetadata" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

Deleting MDS Customizations for adf-config.xml or connections.xml

This section describes how to remove *all* post-deployment configuration for `connections.xml` or `adf-config.xml`. This operation cannot be reversed; customizations are *permanently* removed.

If you **do** want to delete MDS customizations, Oracle recommends that you use the `"exportMetadata"` command to save a copy of the existing files before completing the steps below.

1. Use the `exportMetadata` command to backup `connections.xml` and `adf-config.xml`.

   For example, for WebCenter Portal specify:

```
exportMetadata(application='webcenter', server='WC_Spaces',
 toLocation='/tmp/mydata',
 docs='/META-INF/mdssys/cust/adfshare/adfshare/connections.xml.xml')

exportMetadata(application='webcenter', server='WC_Spaces',
 toLocation='/tmp/mydata',
 docs='/META-INF/mdssys/cust/adfshare/adfshare/adf-config.xml.xml')
```

2. Delete customizations for `connections.xml`, using WLST.

For example, for WebCenter Portal specify:

```
deleteMetadata(application='webcenter', server='WC_Spaces',
 docs='/META-INF/mdssys/cust/adfshare/adfshare/connections.xml.xml')
```

3. Delete customizations for `adf-config.xml`, using WLST.

   For example, for WebCenter Portal specify:

   ```
   deleteMetadata (application='webcenter', server='WC_Spaces',
   docs='/META-INF/mdssys/cust/adfshare/adfshare/adf-config.xml.xml')
   ```

4. Restart the application.
5. Reconfigure your application from scratch using Fusion Middleware Control or WLST.

A.1.2 web.xml

`web.xml` is a standard J2EE application deployment descriptor file and it is located in the `/META-INF` directory for your application. Typical run-time settings in `web.xml` include initialization parameters, custom tag library locations, and security settings.

Most `web.xml` properties are static so they are specified for the application at design time before generating and deploying the application's `.ear` file. If you need to modify some properties in a deployed environment, you can edit some properties through the "Configure Web Modules" screen on the "Deployment Settings" page. See Figure 42–11 in Section 42.1.6.4, "Deploying Applications Using Fusion Middleware Control."

Unlike `connections.xml` and `adf-config.xml`, `web.xml` does *not* store post deployment customizations in MDS and you cannot use Fusion Middleware Control or WLST commands to modify `web.xml` in an existing deployment, such as WebCenter Portal.

Note: Do not edit the `web.xml` file for WebCenter Portal or any Portal Framework application *post deployment*. Oracle does not recommend that you explode application `.ear` files and risk corrupting your installation.

There are very few instances where you might want to modify `web.xml`, for example, in some circumstances you may want to change:

- **Content repository upload parameters**: `UPLOAD_MAX_MEMORY`, `UPLOAD_MAX_DISK_SPACE`, and `UPLOAD_TEMP_DIR`.

  For Portal Framework Applications, see Section 9.12, "Changing the Maximum File Upload Size."

  For WebCenter Portal, use the `uploadedFileMaxDiskSpace` parameter in `webcenter-config.xml` to configure a maximum upload size for files. For details, see Appendix A.1.3, "webcenter-config.xml."

- **Time after which HTTP sessions expire**.

  For Portal Framework Applications, see "Setting HTTP Session Timeout for a Portal Framework Application" in *Oracle Fusion Middleware Performance and Tuning Guide*.

  For WebCenter Portal, see Section 48.10, "Specifying Session Timeout Settings".

- **JSP page timeout value**.

  For Portal Framework Applications, see "Setting JSP Page Timeout" in *Oracle Fusion Middleware Performance and Tuning Guide*.

- **Browser compatibility notifications for Internet Explorer**. Set the `oracle.adf.view.rich.HIDE_UNSUPPORTED_BROWSER_ALERTS` parameter:

```
<!-- Suppress Browser Compatibility popup messages -->
<context-param>
  <param-name>
      oracle.adf.view.rich.HIDE_UNSUPPORTED_BROWSER_ALERTS
  </param-name>
  <param-value>IECompatibilityModes</param-value>
</context-param>
```

  Note: Alternatively, Internet Explorer users can turn off Compatibility Mode before trying to access WebCenter Portal or Portal Framework applications. In Internet Explorer, select the **Tools** menu, and the **Compatibility View Settings**. In the Compatibility View Settings dialog, deselect all the options, and click **Close**.

A.1.3 webcenter-config.xml

`webcenter-config.xml` is a configuration file for the out-of-the-box application WebCenter Portal. This file contains application-level settings, such as the application name and logo. Most of the properties in this file are managed through WebCenter Portal administration screens so there is no need to edit `webcenter-config.xml` directly. For more information, see Chapter 47, "Exploring the Administration Page in Portal Builder Administration" and Chapter 48, "Configuring Global Defaults Across Portals."

There are very few instances where you might be required to manually modify settings in `webcenter-config.xml`; for example, if you want to change the following:

- **Maximum file upload size** (`uploadedFileMaxDiskSpace`) - the default setting is 2 GB.

If you want to modify this setting, you must export the latest version of `webcenter-config.xml` from MDS and modify the `uploadedFileMaxDiskSpace` value as follows:

1. Export the latest `webcenter-config.xml` from MDS.

   For example:

```
exportMetadata(application='webcenter', server='WC_Spaces',
 toLocation='/tmp/mydata',
docs='/oracle/webcenter/webcenterapp/metadata/mdssys/cust/site/webcenter/webcen
ter-config.xml.xml')
```

> **Note:** `webcenter-config.xml` is created in MDS the first time you configure global defaults on the General page in WebCenter Portal Builder administration. If the file does not yet exist in MDS you can edit `webcenter-config.xml` directly. The file is located at: `/oracle/webcenter/webcenterapp/metadata/webcenter-config.xml`

2. Open `webcenter-config.xml.xml` exported from MDS in a text editor and add the following snippet, changing the `uploadedFileMaxDiskSpace` value as required:

```
<mds:replace
node="webcenter(xmlns(webcenter=http://xmlns.oracle.com/webcenter/webcenterapp)
)/webcenter:uploadedFileMaxDiskSpace"/>
<mds:insert
after="webcenter(xmlns(webcenter=http://xmlns.oracle.com/webcenter/webcenterapp
))/webcenter:custom-attributes" parent="webcenter">
<uploadedFileMaxDiskSpace
xmlns="http://xmlns.oracle.com/webcenter/webcenterapp">2147483648</uploadedFile
MaxDiskSpace>
</mds:insert>
```

3. Save and close `webcenter-config.xml.xml`.
4. Import the updated `webcenter-config.xml.xml` file to MDS.

   For example:

```
importMetadata(application='webcenter', server='WC_Spaces',
 fromLocation='/tmp/mydata',
docs='/oracle/webcenter/webcenterapp/metadata/mdssys/cust/site/webcenter/webcen
ter-config.xml.xml')
```

A.2 Cluster Configuration

All post deployment configuration through Fusion Middleware Control, WLST, or the Systems MBean Browser is stored as customizations in the MDS repository. In a cluster environment, since the MDS repository is shared across all nodes, all WebCenter Portal configuration changes done on one node are visible to all nodes in the cluster. To effect configuration changes that are not dynamic, all nodes in the cluster must be restarted. See Section 7.2, "Starting and Stopping Managed Servers for WebCenter Portal Application Deployments."

In WebCenter Portal and Portal Framework applications, most configuration changes that you make through Fusion Middleware Control or using WLST, are not dynamic. For example, when you add or modify connection details for various tools and services (analytics, activity graph, announcements, discussions, documents, events, mail, instant messaging and presence, search, worklists, and so on) you must restart the application's managed server.

There are two exceptions; portlet producer and external application registration is dynamic. Any new portlet producers and external applications that you register are immediately available in your application and any changes that you make to existing connections take effect immediately too.

If you edit configuration files in a cluster environment, then you must ensure that identical changes are made in each cluster member so that the overall cluster configuration remains synchronized.

A.3 Configuration Tools

Oracle offers a range of tools for configuring WebCenter Portal and Portal Framework application deployments. This section outline which tools are available.

Note: Most WebCenter Portal configuration parameters are immutable and cannot be changed at run time unless otherwise specified.

Post deployment, always use Fusion Middleware Control or WebLogic Scripting Tool (WLST) commands to review the latest configuration or make configuration changes. In Fusion Middleware Control you will mostly use WebCenter Portal-specific configuration screens but a useful Systems MBean Browser is also available for reviewing and modifying configuration settings.

For more information about these tools, read:

- Oracle Enterprise Manager Fusion Middleware Control Console
- Oracle WebLogic Scripting Tool (WLST)
- System MBean Browser

These tools always show you the current configuration so, typically, there is no need for you to examine or manually change the content of configuration files or MDS customization data for files such as `adf-config.xml` or `connections.xml`. If you use the same MDS details when you redeploy the application, all configuration performed using these tools is preserved.

What Configuration Tool to Use

You can use any tool for post-deployment configuration. However, if you intend to repeat the configuration steps multiple times, for example, when provisioning newer instances or for automation, screen-based configuration using tools such as Fusion Middleware Control becomes less efficient. In such cases, Oracle highly recommends that you write WLST scripts to perform the required configuration.

All configuration operations possible through Fusion Middleware Control are available using Oracle WebCenter Portal's WLST commands. You can also use WLST scripts to configure other components, for example, to deploy applications, create managed servers, set MDS properties for an application, configure data sources, and so on.

If you want help to automate domain configuration, you can record configuration actions in the WebLogic Server Administration Console as a series of WLST commands and then use WLST to replay the commands. For more details on this topic, see "Recording WLST Scripts" in *Oracle Fusion Middleware Introduction to Oracle WebLogic Server*.

Tip: Where Oracle documentation describes steps in the WebLogic Server Administration Console, consider automating the process using the "Record" option.

Another way to configure deployment specific properties is through the WebCenter Portal or Portal Framework application's deployment plan. Typical properties changed on deployment include:

- Host/port properties for connections
- Standard J2EE artifacts in `web.xml`

See Section 42.1.6, "Deploying the Application to a WebLogic Managed Server."

Note: While reconfiguration is possible this way, any metadata repository and ADF connection configuration changes that you make are not saved as part of the deployment plan, that is, they are saved in the archive that is deployed. Therefore, your configuration changes must be repeated on subsequent redeployments.

If you redeploy your application multiple times, Oracle recommends that you use Fusion Middleware Control or WLST commands to perform your post-deployment configuration. This way, configurations changes are saved in MDS and remain intact on redeployment.

B

Oracle HTTP Server Configuration for WebCenter Portal

When Oracle WebCenter Portal components are running on Oracle WebLogic Server, you can set Oracle HTTP Server (OHS) as the frontend to Oracle WebLogic Server. Some scenarios that require OHS as the frontend are:

- For OSSO to function properly between Site Studio and Oracle Content Server. This is achieved through `mod_osso` of OHS.
- The adequate distribution of load across the Oracle WebLogic Server cluster nodes. This is achieved through `mod_wl` of OHS.
- OHS is required for OAM's WebGate component.
- OHS is used as a reverse proxy.

In these cases, you must configure the `mod_wl_ohs` module to allow requests to be proxied from an OHS to Oracle WebLogic Server.

Sample mod_wl_ohs.conf

After you have configured the `mod_wl_ohs` module using the Fusion Middleware Control, the `mod_wl_ohs.conf` file looks similar to Example B–1. The default location of this file is: *OHS_HOME*`/`*Oracle_WT1*`/instances/`*instance1*`/config/OHS/ohs1/mod_wl_ohs.conf`.

Example B–1 WebCenter Portal - Sample mod_wl_ohs.conf File

```
# WebCenter Portal Application
  <Location /webcenter>
    SetHandler weblogic-handler
    WeblogicHost webcenter.example.com
    WeblogicPort 8888
  </Location>
  <Location /webcenterhelp>
    SetHandler weblogic-handler
    WeblogicHost webcenter.example.com
    WeblogicPort 8888
  </Location>
  <Location /rss>
    SetHandler weblogic-handler
    WeblogicHost webcenter.example.com
    WeblogicPort 8888
  </Location>
  <Location /rest>
    SetHandler weblogic-handler
```

```
    WeblogicHost webcenter.example.com
    WeblogicPort 8888
  </Location>
# Discussions
  <Location /owc_discussions>
    SetHandler weblogic-handler
    WeblogicHost discuss.example.com
    WeblogicPort 8890
  </Location>
# SES Search
  <Location /rsscrawl>
    SetHandler weblogic-handler
    WeblogicHost ses.examplet.com
    WeblogicPort 7777
  </Location>
  <Location /sesUserAuth>
    SetHandler weblogic-handler
    WeblogicHost ses.example.com
    WeblogicPort 7777
  </Location>
# Portlets
  <Location /portalTools>
    SetHandler weblogic-handler
    WeblogicHost webcenter.example.com
    WeblogicPort 8889
  </Location>
  <Location /wsrp-tools>
    SetHandler weblogic-handler
    WeblogicHost webcenter.example.com
    WeblogicPort 8889
  </Location>
  <Location /pagelets>
     SetHandler weblogic-handler
     WeblogicHost webcenter.example.com
     WeblogicPort 8889
   </Location>
# Personalization
  <Location /wcps>
    SetHandler weblogic-handler
    WeblogicHost webcenter.example.com
    WeblogicPort 8891
  </Location>
# Activity Graph
  <Location /activitygraph-engines>
    SetHandler weblogic-handler
    WeblogicHost webcenter.example.com
    WeblogicPort 8891
  </Location>
# UCM
#  Web server context root for Oracle WebCenter Content Server
   <Location /cs>
     SetHandler weblogic-handler
     WeblogicHost ucm.example.com
     WeblogicPort 16200
  </Location>
#  Enables Oracle WebCenter Content Server authentication
   <Location /adfAuthentication>
     SetHandler weblogic-handler
     WeblogicHost ucm.example.com  # Same as /cs entry
     WeblogicPort 16200            # Same as /cs entry
```

```
  </Location>
#  SAML SSO
   <Location /samlacs/acs>
     SetHandler weblogic-handler
     WeblogicHost ucm.example.com
     WeblogicPort 16200
  </Location>
# BPEL Server
  <Location /workflow>
    SetHandler weblogic-handler
    WeblogicHost soa.example.com
    WeblogicPort 8001
  </Location>
# SharePoint
  <Location /wcsdocs>
    SetHandler weblogic-handler
    WeblogicHost webcenter.example.com
    WeblogicPort 8888
  </Location>
  <Location /_vti_bin>
    SetHandler weblogic-handler
    WeblogicHost webcenter.example.com
    WeblogicPort 8888
  </Location>
```

SSL Directives

If you have configured SSL, then the following additional directives are required:

- `WLProxySSL ON`
- `WLProxySSLPassThrough ON`

For example, `mod_wl_ohs.conf` entries looks like Example B–2:

Example B–2 WebCenter Portal - mod_wl_ohs.conf File with SSL Directives

```
# WebCenter Portal Application
  <Location /webcenter>
    SetHandler weblogic-handler
    WeblogicHost webcenter.example.com
    WeblogicPort 8888
       WLProxySSL ON
       WLProxySSLPassThrough ON
  </Location>
  <Location /webcenterhelp>
    SetHandler weblogic-handler
    WeblogicHost webcenter.example.com
    WeblogicPort 8888
       WLProxySSL ON
       WLProxySSLPassThrough ON
  </Location>
...
```

Frontend Listening Host and Frontend Listening Port

If the Oracle HTTP Server (OHS) frontend is also the site entry point, use the Oracle WebLogic Server Administration Console to set the `FrontEnd Host` and `FrontEnd HTTP Port` *for each* server that uses the OHS frontend.

Configuring an Error Page for a Cluster

Include the `ErrorPage` parameter in `mod_wl_ohs.conf` as follows:

```
<Location /webcenter>
 WebLogicCluster
 app1.mycompany.com:8888,app2.mycompany.com:8888,app3.company.com:8888

 SetHandler weblogic-handler
 ErrorPage http://mycompany.com/error.html
 </Location>
```

Users are redirected to `http://company.com/error.html` if all the `WC_Spaces` managed servers are down. When any managed server comes back online, users access WebCenter Portal as normal.

See also, the "Configuring mod_wl_ohs" section in *Oracle Fusion Middleware Administrator's Guide for Oracle HTTP Server*.

C

Third-Party Product Support

This appendix lists third party products that can be used with WebCenter Portal and Portal Framework applications:

Table C–1 WebCenter Portal - Third Party Product Support

Feature	Product and Version	More information
Database	IBM DB2 9.7 and later Microsoft SQL Server 2005 Microsoft SQL Server 2008	See also, System Requirements and Supported Platforms for Oracle Fusion Middleware 11gR1
Identity Store	Supported LDAP Identity Store Types	Section 30.2.3, "Default Identity and Policy Stores"
Documents	Microsoft Office SharePoint Server (MOSS) 2007 SP2	Section 9.3, "Configuring a Microsoft SharePoint Repository"
Documents	Microsoft Windows SharePoint Services (WSS) version 3 SP2	Section 9.3, "Configuring a Microsoft SharePoint Repository"
Events	Microsoft Exchange Server 2007	Section 13.3.1, "Microsoft Exchange Server 2007 Prerequisites"
Events	Microsoft Exchange Server 2003	Section 13.3.2, "Microsoft Exchange Server 2003 Prerequisites"
Mail	Microsoft Exchange Server 2007	Section 15.3.2.1, "Configuring Microsoft Exchange Server 2007 for WebCenter Portal"
Presence	Microsoft Office Live Communications Server (LCS) 2005 R2	Section 14.2.1, "Microsoft Live Communications Server (LCS) Prerequisites"
Presence	Microsoft Office Communications Server (LCS) 2007 SP1	Section 14.2.2, "Microsoft Office Communications Server (OCS) Prerequisites"
Presence	Microsoft Lync 2010	Section 14.2.3, "Microsoft Lync Prerequisites"
Office Integration	Microsoft Office 2007, Microsoft Office 2010, Microsoft Office 2003	Section 13.3.1, "Microsoft Exchange Server 2007 Prerequisites"

D

Oracle Secure Enterprise Search Configuration for Evaluation

This appendix describes how to configure Oracle Secure Enterprise Search (SES) with WebCenter Portal for evaluation purposes. This configuration is not advised for production environments.

This appendix includes the following topics:

- Section D.1, "Understanding the Configuration Script"
- Section D.2, "Configuring an Identity Management System in Oracle SES"
- Section D.3, "Setting Up Oracle WebCenter Content Server for Oracle SES"
- Section D.4, "Setting Up Oracle WebCenter Portal Discussion Server for Oracle SES"
- Section D.5, "Setting Up Oracle SES to Search WebCenter Portal"
- Section D.6, "Running the Configuration Script"

Tip: Read and consider Chapter 18, "Managing Oracle Secure Enterprise Search in WebCenter Portal" to understand the detailed steps required before attempting this simplified configuration.

Permissions: The content of this chapter is intended for WebCenter Portal system administrators. To perform the tasks in this chapter, you must be granted:

- WebLogic Server Admin role through the Oracle WebLogic Server Administration Console.
- WebCenter Portal Administrator role through Portal Builder Administration.
- Portal Framework application Administrator role through the application's Administration Console.

For more information about roles and permissions, see Section 1.8, "Understanding Administrative Operations, Roles, and Tools."

D.1 Understanding the Configuration Script

WebCenter Portal provides a script (`ConfigureSES.py`) that can help configure Oracle SES for evaluation purposes.

Notes:

- This evaluation set-up is for use with environments that are *not* enabled with SSL and SSO.
- The script is supported on Oracle SES **11.1.2.2** and above. Section D.5.5, "Configuring Oracle SES Facets and Sorting Attributes" is applicable only to Oracle SES **11.2.2.2**.

The script performs the following tasks:

- Creates connection to Oracle SES in WebCenter Portal
- Creates Federation Trusted Entity on Oracle SES
- Creates crawl user with crawl role in WebCenter Portal
- Creates Content Server source in Oracle SES
- Creates Discussions source and Announcements source in Oracle SES
- Creates WebCenter source in Oracle SES

The following tasks must be configured before the script is run:

- Section D.2, "Configuring an Identity Management System in Oracle SES"
- Section D.3, "Setting Up Oracle WebCenter Content Server for Oracle SES"
- Section D.4, "Setting Up Oracle WebCenter Portal Discussion Server for Oracle SES"
- Section D.5, "Setting Up Oracle SES to Search WebCenter Portal"

D.2 Configuring an Identity Management System in Oracle SES

This section describes initial configuration in Oracle SES required for searching WebCenter Portal.

1. Log on to Oracle SES with the Oracle SES admin user name and the password specified during installation.
   - Open a browser and enter the URL provided after the Oracle SES installation. (This has the form `http://host:port/search/admin/index.jsp`.)
   - For **Oracle SES 11.2.2.2**, the default admin user name is `searchsys`; however, a different name may be specified during installation.
   - For **Oracle SES 11.1.2.2**, the admin user name is `eqsys`.
2. Oracle SES must be configured with an identity management system to validate and authenticate users. This is necessary for secure searches, so searches return only results that the user is allowed to view based on access privileges.

   Because WebCenter Portal uses identity propagation when communicating with Oracle SES, WebCenter Portal's user base must match that in Oracle SES. One way this can happen is by configuring WebCenter Portal and Oracle SES to the same identity management system, such as Oracle Internet Directory.

Note: For information on all supported identity management systems, see Section 30.2.3, "Default Identity and Policy Stores."

Only one identity plug-in can be set up for each Oracle SES instance. All repositories (Oracle WebCenter Content Server, Oracle WebCenter Portal Discussions Server, and Oracle WebCenter Portal) must share the same user base as Oracle SES.

Oracle SES includes numerous identity plug-ins for identity management systems including Oracle Internet Directory, Oracle WebCenter Content Server, and Microsoft Active Directory. For information, see the Oracle SES documentation included with the product. (This is listed in the WebCenter Portal product area on the Oracle Fusion Middleware documentation library.)

The following example sets up the identity plug-in for Oracle Internet Directory:

a. In the Oracle SES administration tool, navigate to the Global Settings - Identity Management Setup page, select **Oracle Internet Directory** from the available identity plug-ins, and click **Activate**.

b. Provide the following values:

Host name: The host name of the computer where Oracle Internet Directory is running

Port: The Oracle Internet Directory port number

Use SSL: `true` or `false`

Realm: The Oracle Internet Directory realm, for example, `dc=us,dc=oracle,dc=com`

User name: The Oracle Internet Directory admin user name; for example, `cn=orcladmin`

Password: Admin user password

c. Click **Submit**.

3. Each Oracle SES instance must have a trusted entity for allowing WebCenter Portal end users to be securely propagated at search time. (A trusted entity allows the application to authenticate itself to Oracle SES and assert its users when making queries on Oracle SES.) This trusted entity can be any user that either exists on the identity management server behind Oracle SES or is created internally in Oracle SES.

You can create the trusted entity either with the `ConfigureSES.py` script, in WLST, or in Oracle SES.

To have the `ConfigureSES.py` script create the trusted entity, set `ConfiguresSES.properties` with following values:

```
do.config.entity=true
proxy.entity.name=<Federation Trusted Entity name>
prosy.entity.password=<Federation Trusted Entity password>
```

To create the trusted entity with WLST, use the `createFederationTrustedEntity` command. For example:

```
createFederationTrustedEntity(appName='webcenter',
sesUrl='http://mySEShost.com:7777/search/api/admin/AdminServi
```

```
ce', sesPassword='mySESAdminPassword'
entityName='myTrustedEntityUser',
entityPassword='myTrustedEntityUserPassword', desc='Trusted
entity for WebCenter Portal', sesSchema='eqsys')
```

For command syntax and examples, see the "createFederationTrustedEntity" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

To create the trusted entity in Oracle SES, follow these steps.

a. In the Oracle SES administration tool, navigate to the Global Settings - Federation Trusted Entities page.

b. Enter a name for a trusted entity. This is the name that WebCenter Portal uses to authenticate itself to Oracle SES at search time (before it propagates the end user identity to Oracle SES).

To allow the entity to be authenticated through the active identity plug-in:

- Make sure that the entity name is the name of a user that exists in the identity management system.

- Leave the password field blank.

- Select the **Use Identity Plug-in for authentication** check box.

- Enter the authentication attribute value corresponding to the Authentication Attribute in your active identity plug-in.

To allow the entity to be authenticated through Oracle SES:

- Enter any user name (for example, `wcsearch`) and password (for example, `MyPassword1`).

- Do *not* select the **Use Identity Plug-in for authentication** check box.

For more information, see the online help for the Federation Trusted Entities page in Oracle SES.

Note: This trusted entity name and password is required later as the `appUser` and `appPassword` properties in the WLST command `createSESConnection`.

Note: For reference, the following sample user names are used in this chapter:

- `wcsearch`: User of the Oracle SES Federation Trusted Entity
- `mycrawladmin`: Crawl admin user in WebCenter Portal and in the identity management system to crawl certain objects, such as lists, page metadata, portals, and profiles
- `sescrawer` (or admin user): Crawl admin user in Oracle WebCenter Content Server with `sescrawlerrole` (or `admin`) role

---

D.3 Setting Up Oracle WebCenter Content Server for Oracle SES

This section describes how to configure Oracle WebCenter Content Server to be crawlable by Oracle SES (in particular, the Content Server that WebCenter Portal uses for storing documents).

The following steps must be done from within the Content Server.

> **See Also:** Content Server online help for information on administering roles and users in Content Server

1. Create a crawl user.

   If you want users with the `admin` role to crawl, then use an admin user account as the crawl user.

   If you want non-admin users to crawl, then follow these steps:

   a. Create the role `sescrawlerrole`.

   b. Create the user `sescrawler`, and assign it the `sescrawlerrole` role. This user creates the Content Server source in Oracle SES.

   c. Add `sceCrawlerRole=sescrawlerrole` to `config.cfg` (located in *MW_HOME*`/user_projects/domains/`*yourdomain*`/ucm/cs/config`).

      Alternatively, you can append the `sceCrawlerRole=sescrawlerrole` line in the WebCenter Content Server user interface (Administration - General Configuration - Additional Configuration Variables).

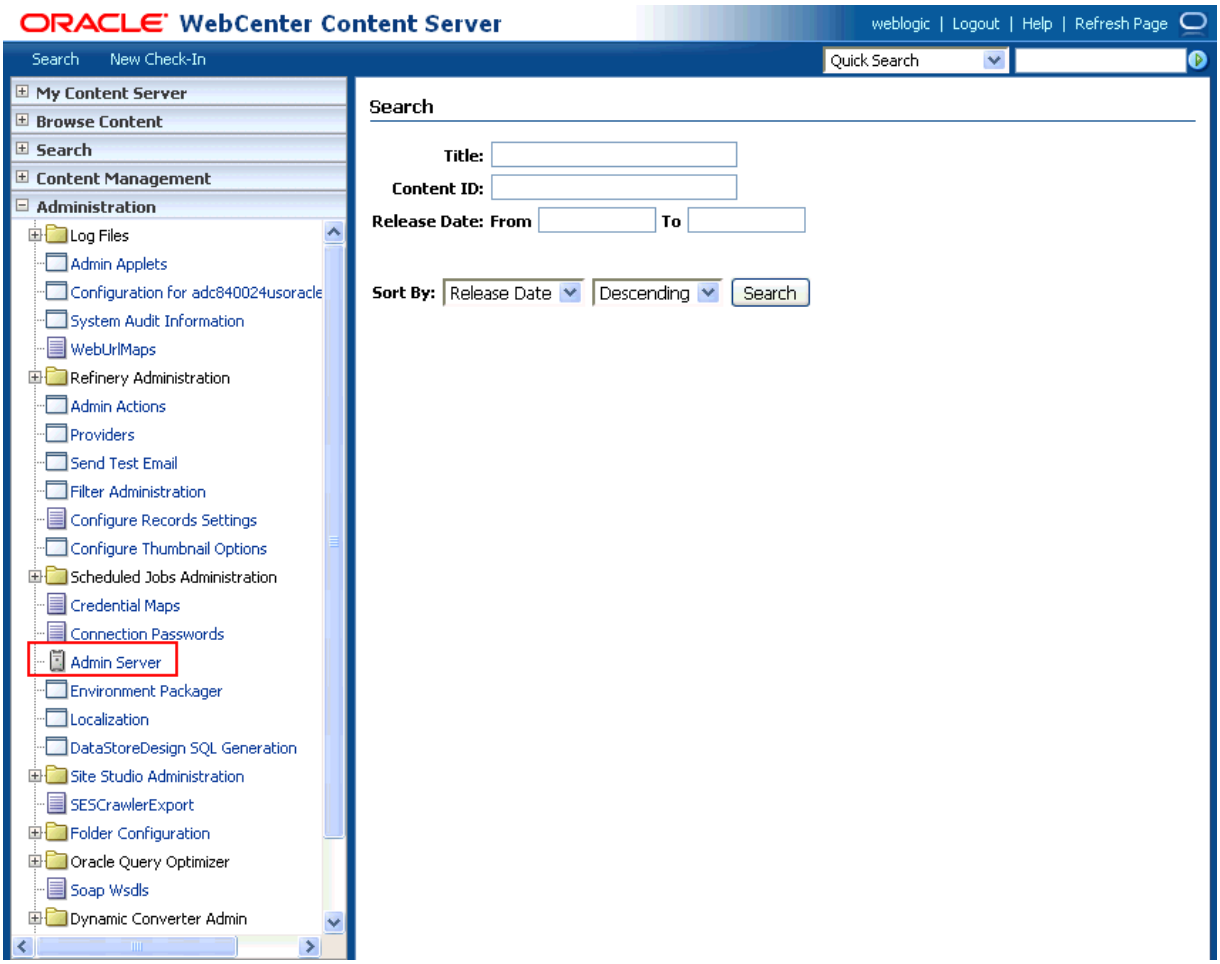
2. Restart the Content Server.

3. In the Content Server console, install the SESCrawlerExport component on the content server, if not done:

   a. Log on to the Content Server as a system administrator. For example: `http://`*host:port*`/cs`.

   b. From the Administration dropdown menu, select **Admin Server** (Figure D–1).

Figure D–1 Content Server Administration



c. Click the button with the instance name.

d. Click **Component Manager** from the menu list on the left pane (Figure D–2).

Figure D–2 Content Server Component Manager

e. Select **SESCrawlerExport** under Integration and click **Update**.

f. Enter configuration parameters. (You can change configuration parameters after installation.)

Disable security on authentication and authorization APIs provided by the SESCrawlerExport; that is, set **Disable Secure APIs** to `false`. This lets

security provided by the SESCrawlerExport be done internally instead of by the content server.

Additionally, in clustered environments only, the **feedLoc** parameter must specify a location on the shared disk accessed by the nodes of the Content Server, and they each must reference it the same way; for example, `sharedDrive/dir1/dir2`. Note that this is not the default location (relative path) provided.

g. Restart the Content Server.

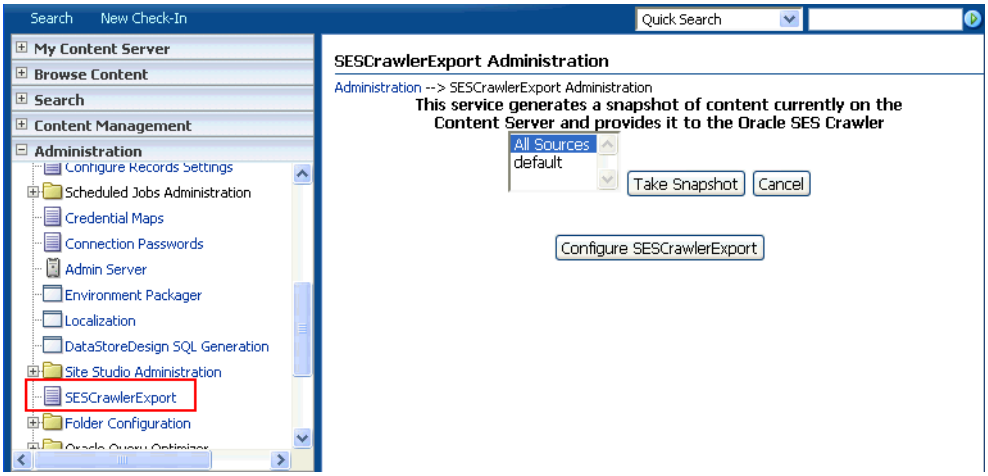
4. Take a snapshot of the Content Server repository.

a. Log on to the Content Server as a system administrator. For example: `http://host:port/cs`.

b. From the Administration dropdown menu, select **SESCrawlerExport**.

c. Select **All sources**, and click **Take Snapshot** (Figure D–3).

Figure D–3 Content Server Snapshot



It is important to take a snapshot before the first crawl or any subsequent full crawl of the source.

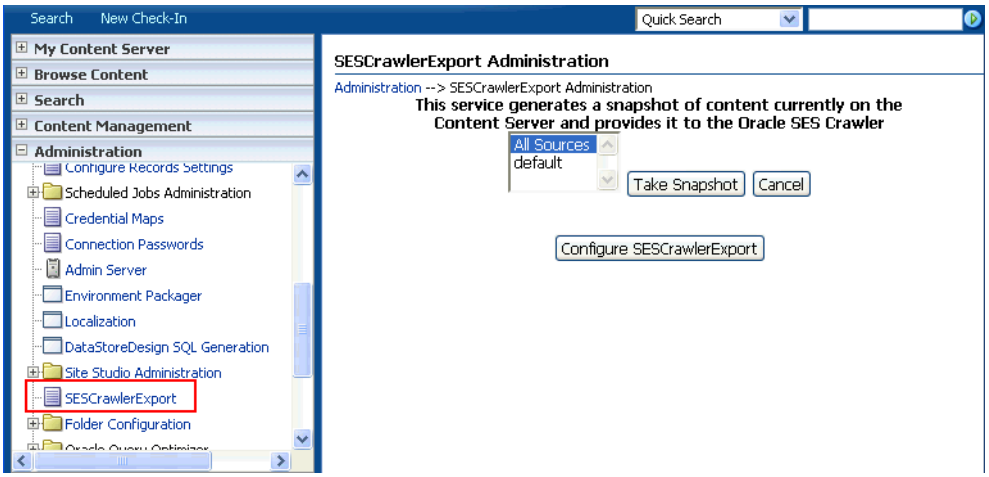
The snapshot generates `configFile.xml` at the location specified during component installation, and feeds are created at the subdirectory with the source name under **feedLoc**.

5. If the Content Server is configured for web rendition, then items in the Content Server are rendered in PDF format. The content item's native MIME type rendition is overwritten. For example, the MIME type of a Microsoft Office Word document is 'application/msword', but when the Content Server uses web rendition the MIME type becomes 'application/pdf'. A search query with the `Mimetype` parameter set to 'application/msword' does not return Word documents.

If your Content Server is configured to use web rendition, then you must configure the Content Server metatdata list to include the `dFormat` value, so that required MIME types are exported to Oracle SES. This is necessary to be able to narrow searches by MIME type.

a. Back on the SESCrawlerExport Administration page, click **Configure SESCrawlerExport** (Figure D–4).

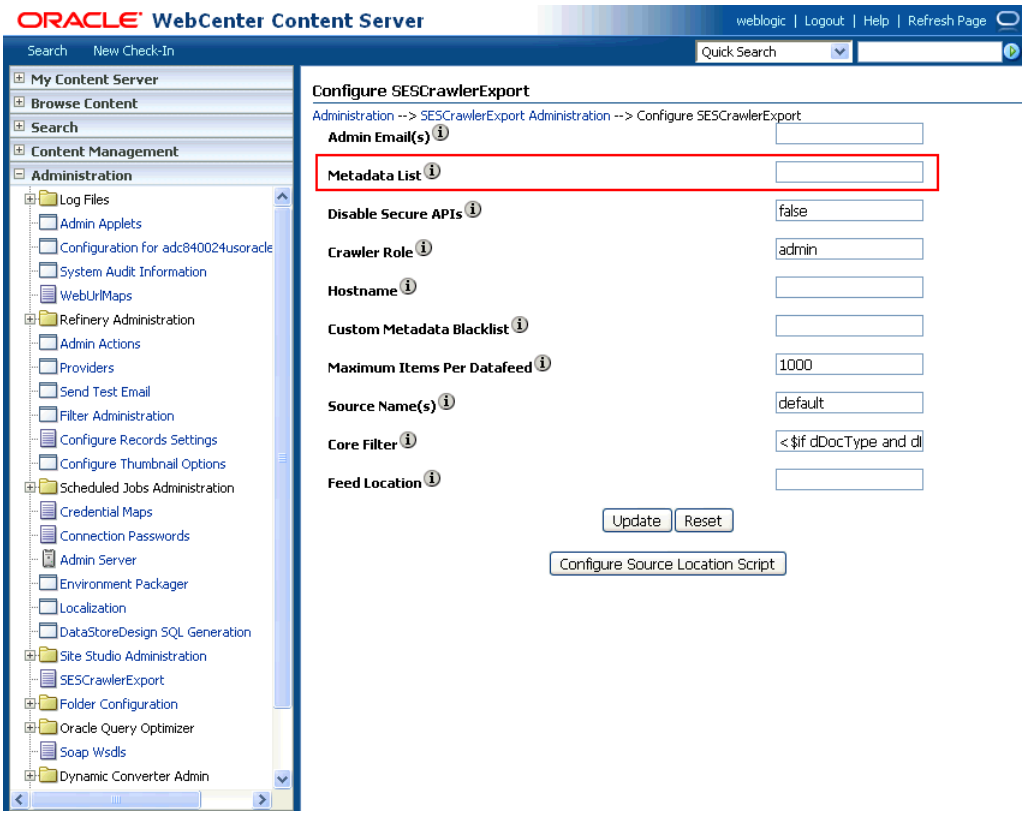
Figure D–4 Content Server Snapshot



b. By default, the **Metadata List** field is blank (Figure D–5). Optionally, add to this field any custom metadata values you require (beginning with `x`). For example, the following entry for **Metadata List** includes custom attributes:

```
xCollectionID, xWCTags, xRegionDefinition
```

Figure D–5 Content Server Metadata List



c. If you are using a version of Content Server *earlier than* 11.1.1.8, then you must add the `dFormat` value to this **Metadata List** field. When the blank default value is changed, the default values are removed, and they must be added

back. (Content Server 11.1.1.8.0 includes most values required for search, including `dFormat`.)

Therefore, if you are using a version of Content Server earlier than 11.1.1.8, you must manually enter the default value for **Metadata List** as follows (plus any custom metadata fields beginning with 'x'):

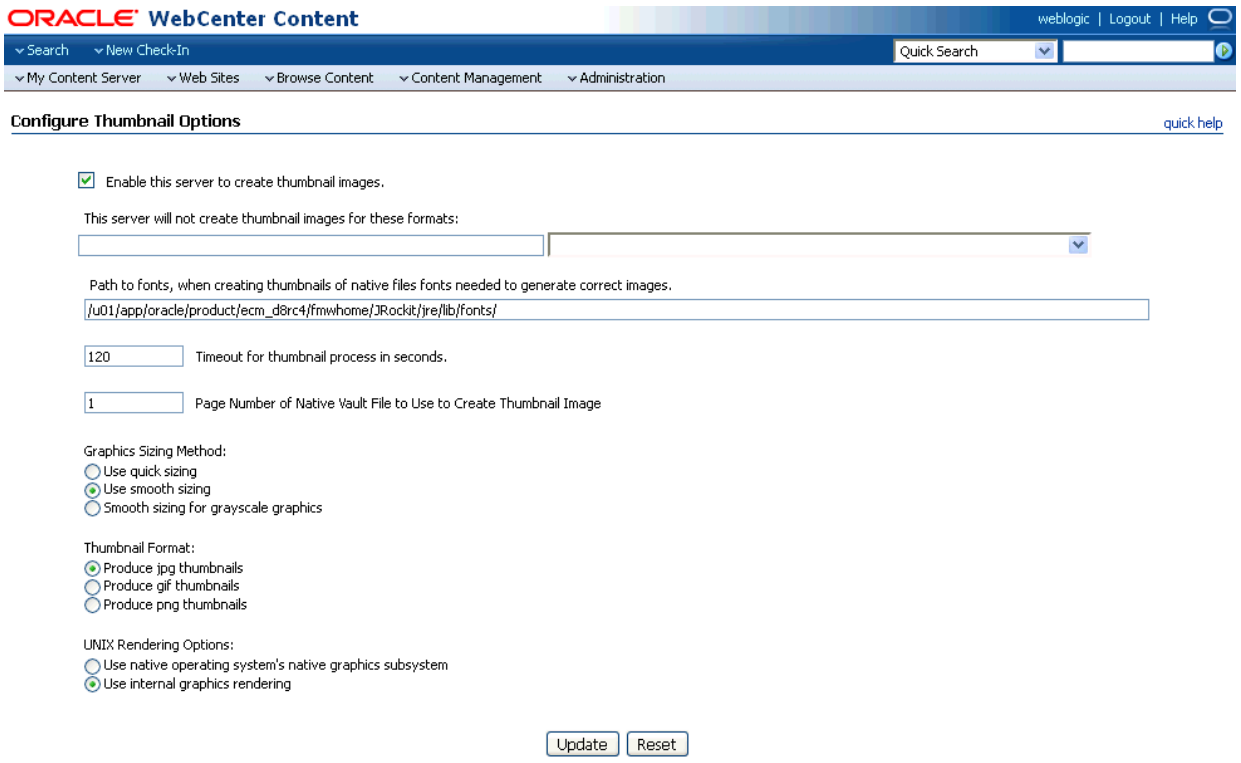
```
dFormat,dID,dDocName,dRevLabel,dDocType,dDocAccount,dSecurityGroup,
dOriginalName,dReleaseDate,dOutDate,dDocCreator,dDocLastModifier,dDocCreate
dDate,dDocFunction
```

6. Configure Thumbnail Options for faceted search.

Note: Oracle SES 11.2.2.2 supports document thumbnails, while earlier releases of Oracle SES do not.

On the **Administration** tab, select **Configure Thumbnail Options** to enable document thumbnails in search results. Leave the default settings as is, and click **Update** (Figure D–6).

Figure D–6 Configure Thumbnail Options



See Also: `Deployment Guide.pdf` included with the product

D.4 Setting Up Oracle WebCenter Portal Discussion Server for Oracle SES

This section describes how to configure Oracle WebCenter Portal's Discussion Server to be crawlable by Oracle SES (in particular, the discussions server that WebCenter Portal uses for storing discussions and announcements).

Note: These steps are not required if you have a new installation of WebCenter Portal (with an Oracle database) and Oracle WebCenter Portal's Discussion Server. It is only required if you are using upgraded (patched) instances.

You can find database schema details for the corresponding data sources from your Oracle WebLogic Server console.

1. Run the Repository Creation Utility (RCU) to confirm that the discussions crawler WebCenter Portal component has been installed on the system.
   - Oracle and Microsoft SQL Server databases:

     Verify that the Oracle WebCenter Portal's Discussion Server back end has been configured properly by noting that the *MyPrefix*_DISCUSSIONS user is installed in RCU.

     Then verify that the discussions crawler has been configured properly by noting that the *MyPrefix*_DISCUSSIONS_CRAWLER user is installed in RCU.
   - IBM DB2 databases:

     Verify that the Oracle WebCenter Portal's Discussion Server back end has been configured properly by noting that the *MyPrefix*_DS user is installed in RCU.

     Then verify that the discussions crawler has been configured properly by noting that the *MyPrefix*_DC user is installed in RCU.

   Note: For IBM DB2 databases, *MyPrefix* is limited to five characters. IBM DB2 uses operating system users for authentication (that is, the database user is actually an operating system user). Because some operating systems have an eight character limit for user names, this requires an eight character user name.

   If the discussions crawler component is not installed, then you must install it using RCU, selecting the same prefix that was used for the Oracle WebCenter Portal's Discussion Server component. Also, during the tablespace specification step in RCU, select *Prefix*_IAS_DISCUSSIONS as the default tablespace. This installs the user for Oracle SES.

   For more information, see Chapter 42, "Deploying Portal Framework Applications."
2. For instances upgraded from WebCenter 11.1.1.1.0 only, run the following tool to upgrade the data in the Oracle WebCenter Portal's Discussion Server database schema, if you have not run the tool yet:

   Note: This step is necessary only if the instance is upgraded from WebCenter 11.1.1.1.0. For instances installed after WebCenter 11.1.1.1.0, this is not required.

```
java -jar \
MW_HOME/discussionserver/discussionserver-upgradeforses.jar \
<command_line_parameters>
```

where *command_line_parameters* are the following MDS schema details and discussions database schema details:

```
-mds_jdbc_user user_id \
-mds_jdbc_password password \
-mds_jdbc_url url \
-discussions_jdbc_user user_id \
-discussions_jdbc_password password \
-discussions_jdbc_url url
```

where *mds_jdbc_user*, *mds_jdbc_password*, and *mds_jdbc_url* are the values to log in to the MDS schema, and *discussions_jdbc_user*, *discussions_jdbc_password*, and *discussions_jdbc_url* are the values to log in to the discussions database schema.

For example:

```
java -jar MW_HOME/as11r1wc/discussionserver/discussionserver-upgradeforses.jar\
-mds_jdbc_user foo \
-mds_jdbc_password MyPassword1 \
-mds_jdbc_url jdbc:oracle:thin:@host:port:SID \
-discussions_jdbc_user foo \
-discussions_jdbc_password MyPassword1 \
-discussions_jdbc_url jdbc:oracle:thin:@host:port:SID
```

D.5 Setting Up Oracle SES to Search WebCenter Portal

The steps in this section must be performed in the Oracle SES administration tool.

The following steps are required:

1. Section D.5.1, "Logging on to the Oracle SES Administration Tool"
2. Section D.5.2, "Setting Up Oracle SES to Search Documents"
3. Section D.5.3, "Setting Up Oracle SES to Search Discussions and Announcements"
4. Section D.5.4, "Excluding Components from the Spaces Crawler"
5. Section D.5.5, "Configuring Oracle SES Facets and Sorting Attributes"
6. Section D.5.6, "Additional Oracle SES Configuration"

> **See Also:** Confirm that you have installed all required patches for Oracle SES. For the latest information on required patches, see the "Back-End Requirements for Search" section in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal* and the Release Notes.
>
> For detailed information about Oracle SES configuration, see the Oracle SES documentation included with the product. (This is listed in the WebCenter Portal product area on the Oracle Fusion Middleware documentation library.)

D.5.1 Logging on to the Oracle SES Administration Tool

Open the Oracle SES administration tool:

1. Open a browser and enter the URL provided after the installation. (This has the form `http://host:port/search/admin/index.jsp`.)
2. Log on with the Oracle SES admin user name and the password specified during installation.

- For **Oracle SES 11.2.2.2**, the default admin user name is `searchsys`; however, a different name may be specified during installation.
- For **Oracle SES 11.1.2.2**, the admin user name is `eqsys`.

D.5.2 Setting Up Oracle SES to Search Documents

To search WebCenter Portal documents using Oracle SES, you must first set up a Document Service Manager (with a Document Service Instance and a Document Service Pipeline).

1. Configure the Document Service Manager (one time for each Oracle SES instance).

Note: Document services are plug-ins involved in the processing of a document when it is being crawled. A document service allows WebCenter Portal to add indexable attributes for documents used in a WebCenter Portal (or Portal Framework) application.

Search attribute names must be unique; two attributes cannot have the same name. For example, if an attribute exists with a String data type, and another attribute is discovered by the crawler with the same name but a different data type, then the crawler ignores the second attribute. Before creating new attributes, make sure to check the list of Oracle SES attribute names and types in the Oracle SES documentation.

---

a. On the Global Settings - Document Services page, click **Create**. Select **Create New Manager**, click **Next**, and enter the following parameters:

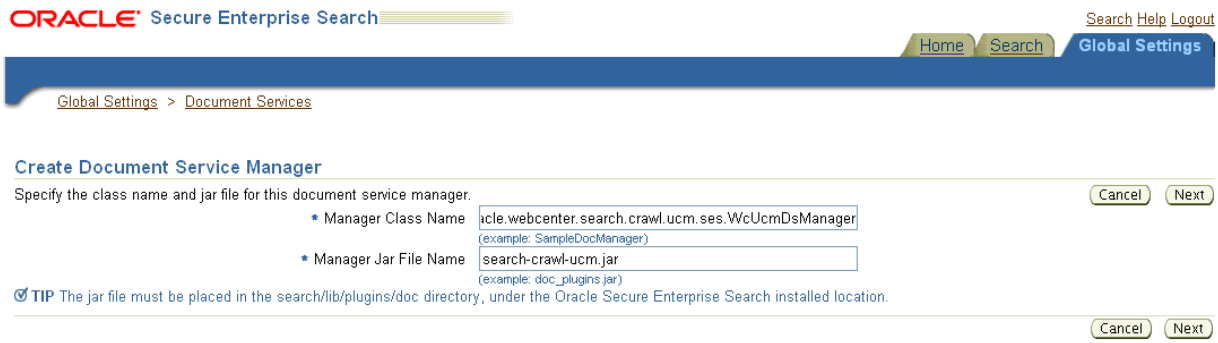
Manager Class Name:
`oracle.webcenter.search.crawl.ucm.ses.WcUcmDsManager`

Manager Jar File Name: `search-crawl-ucm.jar`

Note: The `webcenter_search_ses_plugins.zip` file installs `ORACLE_HOME/search/lib/plugins/doc/search-crawl-ucm.jar`.

Click **Next**, and then click **Finish** (Figure D–7).

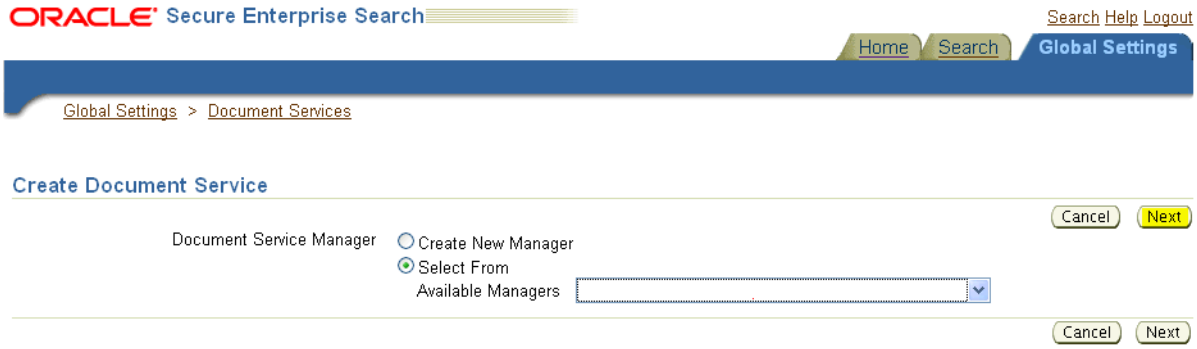
Figure D–7 Creating a Document Service Manager in Oracle SES



b. Create the Document Service Instance.

Again, on the Global Settings - Document Services page, click **Create.** This time, select **Select From Available Managers** with **Secure Enterprise Search WebCenter UCM Plugin**, and click **Next** (Figure D–8).

Figure D–8 Create Document Service



Enter the following parameters:

Instance Name: Enter any name here to be used while creating the document pipeline.

WebCenter Application Name: The unique name being used to identify this application in the back-end Content Server.

Connection Name: The name of the primary Content Server connection that WebCenter Portal (or your Portal Framework application) is using to store documents.

WebCenter URL Prefix: The host and port where the application is deployed; for example: `http://myhost:8888`.

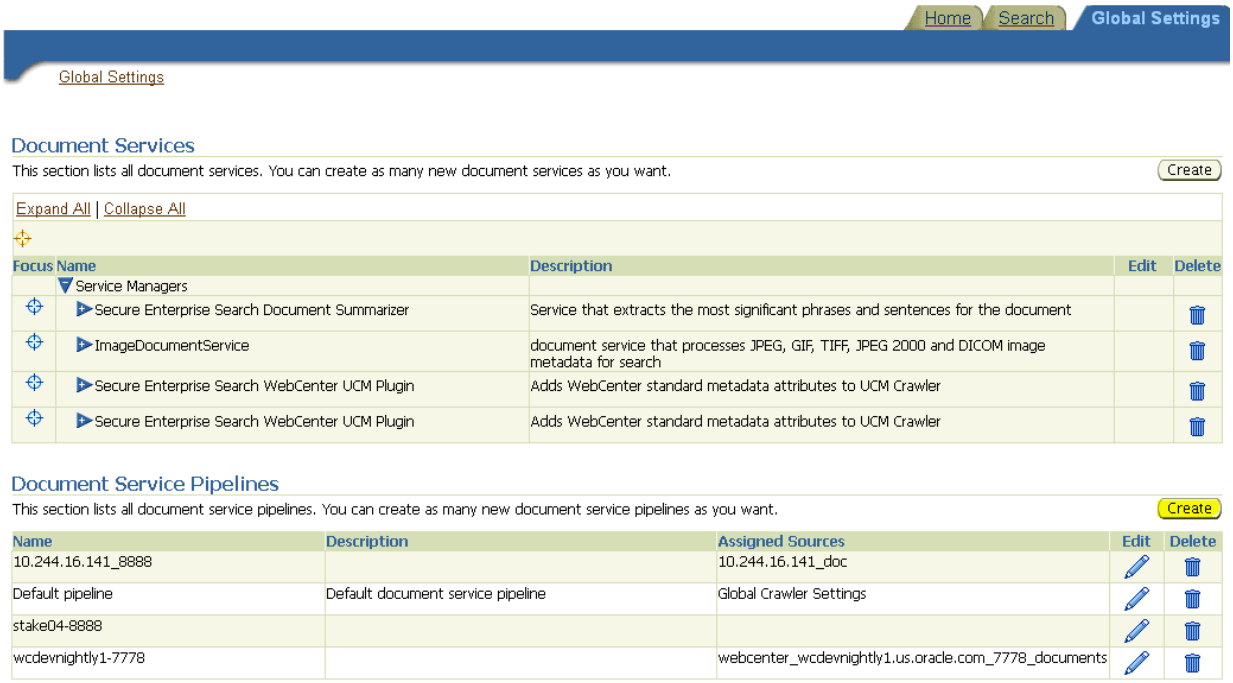
Note: Use Fusion Middleware Control or the `listDocumentsSpacesProperties` command to determine the application name and connection name, as described in Section 18.5.1, "Setting Up WebCenter Portal for Oracle SES."

---

c. Create the Document Service Pipeline. This invokes the document service instance.

Again, on the Global Settings - Document Services page, under the **Document Services Pipelines** section, click **Create** (Figure D–9).

Figure D–9 Creating the Document Service Pipeline



d. On the Create Document Service Pipeline page, enter any name for this pipeline. The document service instance you created in the previous step should be listed under **Available Services**. Select that document service instance, and use the arrow button to move it under **Used in pipeline**.

Note: This pipeline created will be consumed by the `ConfigureSES.py` script to create the document crawler. You will be required to specify this pipeline name for **ucm.pipeline** field in Section D.6, "Running the Configuration Script."

D.5.3 Setting Up Oracle SES to Search Discussions and Announcements

To search WebCenter Portal discussions and announcements using Oracle SES, you must first set up several Oracle SES Database sources: three for discussions and one for announcements. The three discussions sources are for forums, topics in forums, and replies in forums. These separate sources enable users to see search results for forums without also seeing results for all the messages and replies in it.

For example, the discussions sources could have the following:

- source name `GS_Forums` and View of `FORUMCRAWLER_VW`
- source name of `GS_Topics` and View of `THREADCRAWLER_VW`
- source name of `GS_Replies` and View of `MESSAGECRAWLER_VW`

The announcements source could have the source name `GS_Announcements` and a View of `ANNOUNCEMENTS_VW`.

Note: There are slightly different steps for Oracle, Microsoft SQL Server, and IBM DB2 databases.

---

1. Configure the JDBC driver:

   a. To crawl a Microsoft SQL Server or IBM DB2 database, download the appropriate JDBC driver jar files into the *ORACLE_HOME*/search/lib/plugins/oracleapplications directory in Oracle SES.

   Note:

   - For Microsoft SQL Server: Copy the Microsoft JDBC driver files sqljdbc.jar and sqljdbc4.jar.
   - DB2: Copy the IBM driver files db2jcc.jar and db2jcc_license_cu.jar (obtainable from the IBM DB2 UDB client).

   If the JDBC drivers for JRE 1.5 and JRE 1.6 are different, (for example: sqljdbc.jar works for JRE 1.5 and sqljdbc4.jar works for JRE 1.6), then perform the following:

   - Download both the driver jars into the *ORACLE_HOME*/search/lib/plugins/oracleapplications directory in Oracle SES.

   - Add an entry for the JRE 1.6 version (sqljdbc4.jar for SQLServer) of the driver jar to the CLASSPATH element of *ORACLE_HOME*/search/config/searchctl.conf.

   - Restart the middle tier.

   b. Update the drivers.properties file with the following information: *DatabaseName:DriverClassName*.

   c. Add the JRE 1.5 JDBC driver jar file name to the classpath in META-INF/MANIFEST.MF of appsjdbc.jar and DBCrawler.jar.

   For example, change:

   ```
   Class-Path:  sqljdbc.jar rsscrawler.jar  ../../pluginmessages.jar
   ```

   to

   ```
   Class-Path:  db2jcc.jar sqljdbc.jar rsscrawler.jar
   ../../pluginmessages.jar
   ```

   and change:

   ```
   Class-Path: appsjdbc.jar
   ```

   to

   ```
   Class-Path: db2jcc.jar appsjdbc.jar
   ```

   For a key attribute that is not named KEY, change the JDBC driver information in the drivers.properties file to specify the key attribute name: *database_name: driver_class_name, key_attribute_name*

   For example, for a key attribute named ID:

   ```
   oracle : oracle.jdbc.driver.OracleDriver, ID
   ```

In the crawling query, use *key_attribute_name* as the alias for the key value column name. In this example, ID is the alias for KEYVAL:

```
SELECT keyval id, content, url, lastmodifieddate, lang FROM sales_only
```

Oracle and SQL Server databases: The following default drivers are used if none is specified in drivers.properties:

- Oracle: oracle.jdbc.driver.OracleDriver
- SQL Server: com.microsoft.sqlserver.jdbc.SQLServerDriver

2. Required for IBM DB2 databases only:

   a. Make sure that no crawlers are running that use the database crawler source. In the Oracle SES administration tool, check the crawler progress and status on the Home - Schedules page. (Click **Refresh Status**.)

   b. Remake the appsjdbc.jar file and the DBCrawler.jar file. Ensure that the META-INF/MANIFEST.MF was updated correctly; otherwise, the crawler fails with the following error in the crawler log file:

   ```
   EQP-80406: Loading JDBC driver failed
   ```

   c. Modify the Oracle_Home/search/lib/plugins/oracleapplications/drivers.properties file to include the following line:

   ```
   db2: com.ibm.db2.jcc.DB2Driver
   ```

   d. Include the driver jar (db2jcc.jar) to the CLASSPATH element of ORACLE_HOME/search/config/searchctl.conf. For example:

   ```
   #CLASS PATH
   CLASSPATH=ORACLE_HOME/search/webapp/config:ORACLE_HOME/search/webapp/
   SESAuthenticator.jar:ORACLE_HOME/search/lib/plugins/commons-plugins-
   stubs.jar :ORACLE_HOME/search/lib/plugins/oracleapplications/db2jcc.jar
   ```

   e. Edit JVM_OPTIONS in the ORACLE_HOME/search/config/searchctl.conf file to add the system property "-Doracle.home=ORACLE_HOME/search". For example:

   ```
   JVM_OPTIONS= -Djava.awt.headless=true
   -Dweblogic.RootDirectory=ORACLE_HOME/search/base_domain
   -Doracle.home=ORACLE_HOME/search
   ```

   f. Copy the *ORACLE_HOME*/search/lib/plugins/oracleapplications/pluginmessages.jar file to the *ORACLE_HOME*/search/lib directory.

   g. Create the database source. Make sure to enter the correct authorization query and confirm that the attribute name used in **Grant Security Attributes** matches the one used in the authorization query; otherwise, users do not get any results when searching for documents.

D.5.4 Excluding Components from the Spaces Crawler

The spaces crawler collects data for searching the following components:

- oracle.webcenter.peopleconnections.profile (people)
- oracle.webcenter.community (portals)
- oracle.webcenter.page (page metadata)

- `oracle.webcenter.list` (lists)

Use the URL parameter `?excludedServiceIds` to disable search for any of these components. That is, in the Oracle SES administration tool, on the Home - Sources page for the Oracle WebCenter source, the `?excludedServiceIds` in the **Configuration URL** parameter should equal to the comma-delimited list of service IDs to exclude.

Example D–1 Disable Crawling of People Connections Profiles

```
http://host:port/rsscrawl?excludedServiceIds=oracle.webcenter.peopleconnections.profile
```

Example D–2 Disable Crawling of Page Metadata

```
http://host:port/rsscrawl?excludedServiceIds=oracle.webcenter.page
```

Example D–3 Disable Crawling of Profiles and Page Metadata

```
http://host:port/rsscrawl?excludedServiceIds=oracle.webcenter.peopleconnections.profile,oracle.webcenter.page
```

D.5.5 Configuring Oracle SES Facets and Sorting Attributes

Facets are Oracle SES objects that let users refine searches by navigating indexed data without running a new search. You must first define facets (using the provided files) in Oracle SES. Facets defined in Oracle SES are picked up in WebCenter Portal though the **Tools and Services - Search** administration page.

> **Reminder: Oracle SES 11.2.2.2** supports faceted search with WebCenter Portal, but earlier releases of Oracle SES search do not. This section is applicable only to Oracle SES **11.2.2.2**.

WebCenter Portal provides the following input files to the Oracle SES Admin API command line interface:

- `facet.xml`: This configures facets in Oracle SES.
- `searchAttrSortable.xml`: This defines attributes for absolute sort.

Locate these files in `oracle.webcenter.framework/ses/webcenter_portal_ses_admin.zip`. Unzip this file, and follow the instructions in the `readme.txt` file.

Running these two files from Oracle SES creates the following facets:

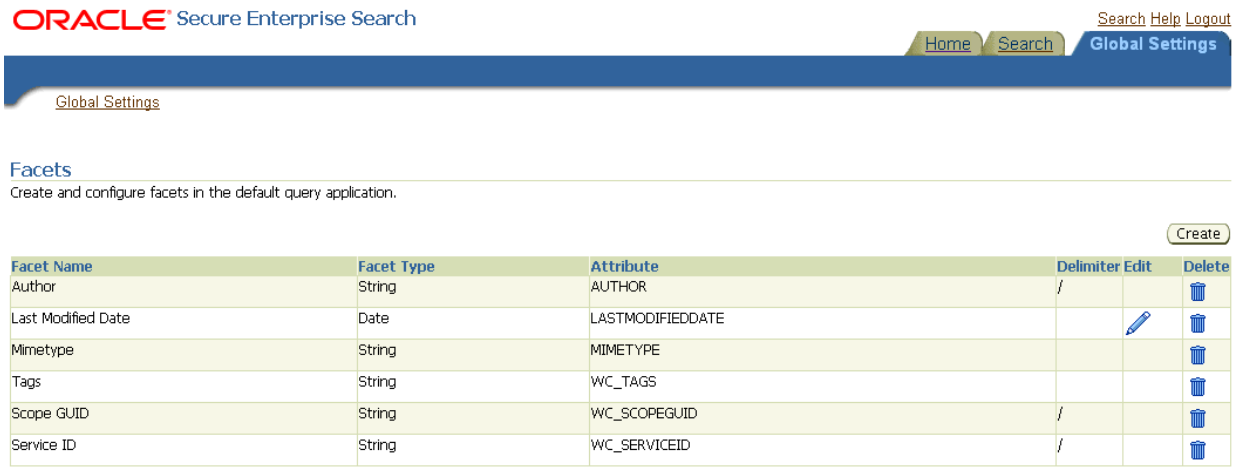
- `Author`
- `Last Modified Date`
- `Mimetype`
- `Tags`
- `Scope GUID` (This appears as the **Portal** facet. This value is converted to the portal display name in the search results page.)
- `Service ID` (This facet does not appear in the user interface. All enabled tools and services display in the search results page.)

Notes: The `facet.xml` and `searchAttrSortable.xml` scripts are mandatory. Creating facets in Oracle SES alone is not sufficient for search in WebCenter Portal.

Additionally, the `Scope GUID` and the `Service ID` facets are mandatory. Facet names are case-sensitive. You must have these exact facet names.

After you run these files, you view facets in the Oracle SES administration tool on the Global Settings - Facets page (Figure D–10).

Figure D–10 Oracle SES Facets



To create a new facet, on the Global Settings - Facets page, click **Create**. Enter a name for the facet and the search attribute from which the facet value should be generated. For String facet types, you must also enter the path delimiter. This is a single character used for demarcation for displaying the facet tree hierarchy for the selected facet tree node on the query page, for example, "tools/power tool/drills", where "/" is the path delimiter. You can set it to blank if the facet tree is one-level deep; that is, its nodes do not have child nodes.

Click **Create and Customize** to create a facet and configure its nodes in the Edit Facet screen. You can configure facet nodes for a facet of Date type or Number type. For example, for the Last Modified Date facet, you can create nodes like Last Year, Last Month, Today, Between two specific days, and so on.

The Node Configuration tab displays a facet hierarchy in tree format as well as in XML format, where you can add, edit, and delete child nodes for the selected facet node. After editing the facet nodes, click **Apply** to save the changes.

Note: Do not modify or delete the `Scope GUID` or `Service ID` facets.

Changes you make in Oracle SES are picked up in WebCenter Portal when the application specialist goes to the **Tools and Services - Search** administration page. WebCenter Portal does not detect changes to facets until this Search administration page is opened. WebCenter Portal remembers the facets selected for use by each portal.

D.5.6 Additional Oracle SES Configuration

Optionally, you can configure the security filter lifespan. This refreshes the authorization policies for users in the system. It is best to have a short lifespan when user policies change frequently. (This appendix uses Oracle Internet Directory identity plug-in as the example.)

For example, in the Oracle SES administration tool, on the Global Settings - Query Configuration page under **Secure Search Configuration**, enter `0` for **Security Filter Lifespan (minutes)**.

Valid values for the security filter lifespan are between 0 minutes (no cache) and 526500 minutes (cache for one year).

Note: The ConfigureSES.py runs the `setSESVersion` WLST command, so that command is not required as part of this chapter's configuration.

D.6 Running the Configuration Script

Oracle provides the `ConfigureSES.py` Python script to simplify the remaining Oracle SES configuration. This automated configuration includes the following:

- Creates connection to Oracle SES in WebCenter Portal
- Creates Federation Trusted Entity on Oracle SES
- Creates crawl user with crawl role in WebCenter Portal
- Creates Content Server source in Oracle SES
- Creates Discussions source and Announcements source in Oracle SES
- Creates WebCenter source in Oracle SES

The script and its properties file are in the *WCP_ORACLE_HOME*`/webcenter/scripts/ses_11.2.2.2/` directory.

Follow these steps to run the script:

1. Set an environment variable to reference the directory. For example:

```
setenv SESDIR Oracle_WC1/webcenter/scripts/ses_11.1.2.2/
```

2. Update the `ConfigureSES.properties` file with appropriate values. Example D–4 shows sample values for a WebCenter Portal application.

Example D–4 WebCenter Portal Application

```
# Required update:
#   Actions can be selected or deselected. See properties with 'do' prefix.
#   All 'do*' properties have 'false' value.
#   Change value to 'true' after meeting prerequisite by filling properties with
 no values.
#   For properties with values, check that they are valid in your environment.
#
# Preset values:
#   Documents crawler and Jive database crawler sections contain values of common
 repository used by WebCenter development team.

# ---------------------------------
```

```
# Prerequisite - wls instance must be started
admin.user=weblogic
admin.url=app_server.example.com:7005
admin.server=app_server.example.com
admin.password=welcome1
webcenter.host=app_server.example.com
webcenter.port=8892

# Prerequisite - SES instance must be started
ses.host=ses_server.example.com
ses.port=5720
ses.schema=searchsys
ses.admin.password=welcome1

# Prerequisite - Entity name must not exist
do.config.entity=true
proxy.entity.name=webcenterproxy
proxy.entity.password=welcome1

# Prerequisite - webcenter-wls instance must be started
do.config.spaces.crawl.user=true
do.config.connection=true
do.spaces.crawl=true
spaces.crawl.password=welcome1
spaces.crawl.authuseridformat=nickname

# Prerequisite - UCM instance must be started
do.document.crawl=true
ucm.host=ucm_server.example.com
ucm.port=9400
ucm.source=default
ucm.admin.user=weblogic
ucm.admin.password=welcome1
ucm.crawl.authuseridformat=nickname

# Prerequisite - Documents Instance and Documents Pipeline on SES must be created
ucm.pipeline=stake04-8888

# Prerequisite - Jive database must be started
do.discussions.crawl=false
discussions.host=discussions_server.example.com
discussions.port=8890
discussions.db.connstring=jdbc:oracle:thin:@wcdb.example.com:1521/wcdb
discussions.db.admin.user=WC_DISCUSSIONS_CRAWLER
discussions.db.admin.password=welcome1
discussions.crawl.authuseridformat=nickname
```

3. Set the default directory to the directory of `wlst.sh` script. For example:

   ```
   cd Oracle_WC1/oracle/as11gr1wc/common/bin/
   ```

4. Run the `ConfigureSES.py` script with `ConfigureSES.properties`. For example:

   ```
   ./wlst.sh $SESDIR/ConfigureSES.py $SESDIR/ConfigureSES.properties
   ```

5. Upon successful completion of the script, you must restart the managed server on which the application is deployed (by default, WC_Spaces). For more information, see the "Starting and Stopping Managed Servers Using WLST" section in *Oracle Fusion Middleware Administrator's Guide*.

E

Labeling During WebCenter Portal Lifecycle

This appendix describes how to use internal labels to keep portal instances, such as stage and production portals, in-sync. You can only propagate portal changes to another portal instance when their internal labels match.

Every portal deployment maintains an internal label. Whenever you deploy or propagate a portal to another server, the source portal's label is copied to the target along with the portal. Similarly, if you deploy a portal to an archive (for export or back up purposes), the portal's label is included within the `.par` file.

Table E–1 illustrates a labeling scenario where a **Sales** portal is deployed from stage to production, followed by subsequent propagations and redeployments between stage and production.

> **Note:** Portal archive exports contain a label with the pattern PTL_ or PTLEXPPORTAL_ depending on which phases a portal goes through. Portals that undergo at least one deployment contain the PTL_ label pattern. Portals that are not yet deployed (when exported) contain the PTLEXPPORTAL_ label pattern.

Table E–1 Internal Labeling on Portal Deployment and Propagation

Step	Action	Labels in Stage	Labels in Production
1	Deploy the Sales portal on stage to production Use WLST command `deployWebCenterPortal`	**PTL_GUID_#$#_1_user** PTLEXPPORTAL_GUID_#$#_1_user	**PTL_GUID_#$#_1_user**
2	Change the Sales portal on the stage and propagate changes to production Use WLST command `propagateWebCenterPortal`	PTL_GUID_#$#_1_user PTLEXPPORTAL_GUID_#$#_1_user **PTL_GUID_#$#_2_user**	PTL_GUID_#$#_1_user - **PTL_GUID_#$#_2_user**
3	Redeploy the Sales portal on stage to production Use WLST command `deployWebCenterPortal`	PTL_GUID_#$#_1_user PTLEXPPORTAL_GUID_#$#_1_user PTL_GUID_#$#_2_user **PTL_GUID_#$#_3_user** PTLEXPPORTAL_GUID_#$#_2_user	PTL_GUID_#$#_1_user - PTL_GUID_#$#_2_user **PTL_GUID_#$#_3_user** -

Table E–1 (Cont.) Internal Labeling on Portal Deployment and Propagation

Step	Action	Labels in Stage	Labels in Production
4	Propagate Sales portal changes on stage to production Use WLST command `propagateWebCenterPortal`	PTL_GUID_#$#_1_user	PTL_GUID_#$#_1_user
		PTLEXPPORTAL_GUID_#$#_1_user	-
		PTL_GUID_#$#_2_user	PTL_GUID_#$#_2_user
		PTL_GUID_#$#_3_user	PTL_GUID_#$#_3_user
		PTLEXPPORTAL_GUID_#$#_2_user	-
		PTL_GUID_#$#_4_user	**PTL_GUID_#$#_4_user**
5	Back up the production portal Use WLST command `exportWebCenterPortals`	PTL_GUID_#$#_1_user	PTL_GUID_#$#_1_user
		PTLEXPPORTAL_GUID_#$#_1_user	-
		PTL_GUID_#$#_2_user	PTL_GUID_#$#_2_user
		PTL_GUID_#$#_3_user	PTL_GUID_#$#_3_user
		PTLEXPPORTAL_GUID_#$#_2_user	-
		PTL_GUID_#$#_4_user	**PTL_GUID_#$#_4_user**
		-	PTLEXPPORTAL_GUID_#$#_1_user
6	Back up the stage portal Use WLST command `exportWebCenterPortals`	PTL_GUID_#$#_1_user	PTL_GUID_#$#_1_user
		PTLEXPPORTAL_GUID_#$#_1_user	-
		PTL_GUID_#$#_2_user	PTL_GUID_#$#_2_user
		PTL_GUID_#$#_3_user	PTL_GUID_#$#_3_user
		PTLEXPPORTAL_GUID_#$#_2_user	-
		PTL_GUID_#$#_4_user	**PTL_GUID_#$#_4_user**
		-	PTLEXPPORTAL_GUID_#$#_1_user
		PTLEXPPORTAL_GUID_#$#_3_user	-
7	Propagate Sales portal changes on stage to production Use WLST command `propagateWebCenterPortal`	PTL_GUID_#$#_1_user	PTL_GUID_#$#_1_user
		PTLEXPPORTAL_GUID_#$#_1_user	-
		PTL_GUID_#$#_2_user	PTL_GUID_#$#_2_user
		PTL_GUID_#$#_3_user	PTL_GUID_#$#_3_user
		PTLEXPPORTAL_GUID_#$#_2_user	-
		PTL_GUID_#$#_4_user	PTL_GUID_#$#_4_user
		-	PTLEXPPORTAL_GUID_#$#_1_user
		PTLEXPPORTAL_GUID_#$#_3_user	-
		PTL_GUID_#$#_5_user	**PTL_GUID_#$#_5_user**

Table E–1 (Cont.) Internal Labeling on Portal Deployment and Propagation

Step	Action	Labels in Stage	Labels in Production
8	Corruption in production so restore the production Sales portal from the latest backup archive (created in step 5) Use WLST command `importWebCenterPortals`	PTL_GUID_#$#_1_user	PTL_GUID_#$#_1_user
		PTLEXPPORTAL_GUID_#$#_1_user	-
		PTL_GUID_#$#_2_user	PTL_GUID_#$#_2_user
		PTL_GUID_#$#_3_user	PTL_GUID_#$#_3_user
		PTLEXPPORTAL_GUID_#$#_2_user	-
		PTL_GUID_#$#_4_user	PTLEXPPORTAL_GUID_#$#_1_user
		-	-
		PTLEXPPORTAL_GUID_#$#_3_user	-
		PTL_GUID_#$#_5_user	-
		-	**PTL_GUID_#$#_4_user**
9	Propagate Sales portal changes on stage to production Use WLST command `propagateWebCenterPortal`	PTL_GUID_#$#_1_user	PTL_GUID_#$#_1_user
		PTLEXPPORTAL_GUID_#$#_1_user	-
		PTL_GUID_#$#_2_user	PTL_GUID_#$#_2_user
		PTL_GUID_#$#_3_user	PTL_GUID_#$#_3_user
		PTLEXPPORTAL_GUID_#$#_2_user	-
		PTL_GUID_#$#_4_user	PTLEXPPORTAL_GUID_#$#_1_user
		-	-
		PTLEXPPORTAL_GUID_#$#_3_user	-
		PTL_GUID_#$#_5_user	-
		-	PTL_GUID_#$#_4_user
		PTL_GUID_#$#_6_user	**PTL_GUID_#$#_6_user**
10	Corruption in the stage, so restore the stage Sales portal from the latest backup archive (created in step 6) Use WLST command `importWebCenterPortals`	PTL_GUID_#$#_1_user	PTL_GUID_#$#_1_user
		PTLEXPPORTAL_GUID_#$#_1_user	-
		PTL_GUID_#$#_2_user	PTL_GUID_#$#_2_user
		PTL_GUID_#$#_3_user	PTL_GUID_#$#_3_user
		PTLEXPPORTAL_GUID_#$#_2_user	-
		PTLEXPPORTAL_GUID_#$#_3_user	PTLEXPPORTAL_GUID_#$#_1_user
		-	-
		-	-
		-	-
		-	PTL_GUID_#$#_4_user
		-	**PTL_GUID_#$#_6_user**
		P**TL_GUID_#$#_4_user**	-
11	Propagate Sales portal on stage to production	Propagation fails because the portal on production is newer than the version on stage. Either redeploy the Sales portal or restore the production portal on stage (as per steps 12 and 13).	-

Table E–1 (Cont.) Internal Labeling on Portal Deployment and Propagation

Step	Action	Labels in Stage	Labels in Production
12	Export the Sales portal from production	PTL_GUID_#$#_1_user	PTL_GUID_#$#_1_user
		PTLEXPPORTAL_GUID_#$#_1_user	-
		PTL_GUID_#$#_2_user	PTL_GUID_#$#_2_user
		PTL_GUID_#$#_3_user	PTL_GUID_#$#_3_user
		PTLEXPPORTAL_GUID_#$#_2_user	-
		PTLEXPPORTAL_GUID_#$#_3_user	PTLEXPPORTAL_GUID_#$#_1_user
		-	-
		-	-
		-	-
		-	PTL_GUID_#$#_4_user
		-	**PTL_GUID_#$#_6_user**
		PTL_GUID_#$#_4_user	-
			PTLEXPPORTAL_GUID_#$#_2_user
13	Restore the Sales portal on stage from the Sales portal archive exported from production (in step 12)	PTL_GUID_#$#_1_user	PTL_GUID_#$#_1_user
		PTLEXPPORTAL_GUID_#$#_1_user	-
		PTL_GUID_#$#_2_user	PTL_GUID_#$#_2_user
		PTL_GUID_#$#_3_user	PTL_GUID_#$#_3_user
		PTLEXPPORTAL_GUID_#$#_2_user	-
		PTLEXPPORTAL_GUID_#$#_3_user	PTLEXPPORTAL_GUID_#$#_1_user
		-	-
		-	-
		-	-
		-	PTL_GUID_#$#_4_user
		-	**PTL_GUID_#$#_6_user**
		PTL_GUID_#$#_4_user	-
		-	PTLEXPPORTAL_GUID_#$#_2_user
		PTL_GUID_#$#_6_user	-

The labels are for internal use only so there is no need for you to view or manage these labels. If there is a mismatch between the source and target labels an error message displays. For example:

Scenario 1: You attempt to propagate a portal that was not previously deployed on the target. The following message displays because the portal's initial deployment label is missing from the target:

```
Internal label for deployment for the portal <portal_name> does
not exist on the target. Ensure that the portal is deployed on
the target.
```

Scenario 2: You attempt to propagate a portal but the label on the target does not exist on the source. The following message displays because the label in the source is lower than the target label:

```
Cannot propagate the portal. Internal labels in the source and
target for portal {0} do not match. Redeploy the portal on the
target to synchronize the portals before attempting further
propagation.
```

The type of mismatch can occur if the source portal was restored with an earlier label than the target.

F

Migrating Wiki Content to WebCenter Portal

This appendix describes how to migrate wiki content from wiki applications, such as Confluence, into WebCenter Portal using a custom wiki extraction tool in combination with the Document Migration Utility. The custom wiki extraction tool extracts the wiki content into an archive format that you can import into a WebCenter Portal content repository, using the Document Migration Utility. You should have an understanding of the content created in Content Server for a portal, portal template, wiki documents, and wiki pages, and a detailed understanding of the Document Migration Utility and the format of its archive.

> **Note:** In this WebCenter Portal release, do not use the Document Migration Utility to export or import portal folders and portal template folders. Starting in release 11.1.1.8.0, content folders are exported with the portal or portal template. For details, see Section 40.1, "Deploying Portals" and Section 40.2, "Deploying Portal Templates."

This appendix includes the following topics:

- Section F.1, "Understanding Wiki Documents and Wiki Pages"
- Section F.2, "Understanding the Document Migration Utility"
- Section F.3, "Migrating Data from the Source Wiki Application to WebCenter Portal"

F.1 Understanding Wiki Documents and Wiki Pages

This section describes the format and how wiki documents and wiki pages work in WebCenter Portal. For more information about wiki documents and wiki pages in WebCenter Portal, see the "Working with Wikis" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal*.

This section contains the following topics:

- Section F.1.1, "Understanding Wiki Documents"
- Section F.1.2, "Understanding Wiki Pages"

F.1.1 Understanding Wiki Documents

In WebCenter Portal you can create a wiki document using document task flows. These documents can reside anywhere in the hierarchy of any created folders inside a

portal. Wiki documents can sit alongside documents of other types, or you could choose to arrange all your wiki documents inside a single folder.

When a wiki document is created in WebCenter Portal, an HTML document is created and checked into Content Server. This wiki document contains special metadata values that tell WebCenter Portal that the document is a wiki document as opposed to a regular HTML document. These metadata values are:

```
dDocType = Application
dDocFunction = wiki
dOriginalName (document filename) = <wikiName>.htm
```

When you open a document in WebCenter Portal with the above metadata, WebCenter Portal will know to display it as a wiki document.

F.1.2 Understanding Wiki Pages

In WebCenter Portal you can create a wiki page by creating a page based on the `Wiki` page template. When you navigate to a wiki page you are presented with a wiki document. Depending on the display template of your portal, you may see a **Wikis** menu, which you can click to list all the wiki pages in your portal. See the "Working with Wikis" chapter in *Oracle Fusion Middleware Using Oracle WebCenter Portal* for details on how to create wiki pages.

When a wiki page is created in WebCenter Portal the following artifacts are created in Content Server:

- A folder is created in the folder for the portal the wiki page is being created in; the name of the wiki folder is the same name as the wiki page name but with special characters removed.
- A document is created inside the wiki folder with the following metadata:
  - `dDocTitle` = document title (same name as the wiki page name with an extension of `.htm`)
  - `dOriginalName` = the documents filename (same as dDocTitle)
  - `dDocFunction` = `wiki`
  - `dDocType` = `Application`
  - `xWCPageID` = the name of the wiki page's JSPX page

This is best illustrated with an example. If the root folder is `WebCenter`, the portal in which a wiki page is being created is `Marketing`, and the wiki page being created is `Wiki1`, the following artifacts will be created in Content Server:

- Folder: `/WebCenter/Marketing/Wiki1`
- Document: `/WebCenter/Marketing/Wiki1/Wiki1.htm`
  - `dDocTitle= Wiki1.htm`
  - `dOriginalName = Wiki1.htm`
  - `dDocFunction = 'wiki'`
  - `dDocType = 'Application'`
  - `xWCPageID = Wiki1.jspx`

When you navigate to a wiki page the following occurs:

- WebCenter Content is queried for the document in the following location:

`/<RootFolder>/<PortalFolder>/<wikiPageName>/<wikiPageName>.htm`

For example: `/WebCenter/Marketing/Wiki1/Wiki1.htm`

- If the document is found, it is displayed as a wiki document.
- If the document is not found the wiki page will display the contents of the wiki folder.

F.2 Understanding the Document Migration Utility

In this WebCenter Portal release (11.1.1.8.3), only use the Document Migration Utility to migrate content from an existing wiki application to WebCenter Portal as described in Section F.3, "Migrating Data from the Source Wiki Application to WebCenter Portal."

Note: Do not use the Document Migration Utility to export or import portal folders and portal template folders. In this release, content folders are exported with the portal or portal template. For details, see Section 40.1, "Deploying Portals" and Section 40.2, "Deploying Portal Templates."

This section describes the Document Migration Utility. It contains the following topics:

- Section F.2.1, "Understanding the Document Migration Utility's Export Function"
- Section F.2.2, "Understanding the Document Migration Utility's Import Function"

F.2.1 Understanding the Document Migration Utility's Export Function

On export the document content stored in Content Server for the specified portals and portal templates are extracted into a Document Migration Archive. The Document Migration Archive extracts all the contents of a portal or portal template into its own top-level folder in the archive. The name of this folder is based on the internal GUID of the portal or portal template folder in Content Server.

The Document Migration Archive will contain a root folder for each portal or portal template in the export. That is, if you are exporting content from four portals, the archive will have four root folders. The names of these root folders is based on the internal GUID of the corresponding portal or portal template folder in Content Server.

In order to have one archive per portal or portal template, the Document Migration Utility should be called specifying one portal/portal template at a time and a unique archive name.

Inside each top level folder in the archive is an XML document (`ExportImportData.xml`) that fully describes the data in that root folder (that is, it describes the data exported for that portal or portal template). It contains all the metadata of the folders and files in the source Content Server that are to be maintained on import. For more information about the format of the archive, see Section F.2, "Understanding the Document Migration Utility." For more information about the metadata maintained between the source and target, see Section F.2.2.1, "Understanding How the Document Migration Utility Handles Metadata."

Assuming there are portals or portal templates for which contents are to be exported, the following occurs during export:

- A temporary directory is created.

- For each portal or portal template:
  - a directory is created in the temporary directory with its name being the GUID of the corresponding folder in Content Server
  - an `ExportImportData.xml` document is created inside that new directory
  - Content Server is queried for the contents of the portal/portal template
  - If there are contents, they are read and information about the folders/documents are written to the `ExportImportData.xml` document and the files are downloaded into the temporary directory
  - If there are contents, they are read and information about the folders and documents are written to the `ExportImportData.xml` document. The actual files are downloaded into the directory maintaining the folder hierarchy as in the Content Server. For example, if in the Content Server the document `doc1.doc` is in folder `Folder1` the document will be added to the root folder under `Folder1` as shown below.

    ```
    /<tmpDir>/<portalFolder>/Folder1/doc1.doc
    ```

- After all contents have been extracted, the entire contents of the temporary directory are zipped up into the Document Migration Archive.

F.2.2 Understanding the Document Migration Utility's Import Function

Before you import content using the Document Migration Utility, the target portal and/or portal templates must exist on the target Content Server. If these folders do not exist in Content Server the import of document content will fail.

> **Note:** You cannot simply create a portal or portal template in the target with the same name as in the source and expect it to work. If you do this, the folder in Content Server will have the same name as in the source but with a different GUID. It is this folder GUID that is used to tie the portal in WebCenter Portal with the folder in Content Server as well as what is used to synchronize the data in the export archive with the target folder in the target Content Server.

On import, the Document Migration Archive is read and for each portal or portal template's data the content is recreated in the corresponding portal or portal template's folder in the target Content Server. The names of the top level folders in the archive are used to synchronize with the portal or portal template folder in the target Content Server instance. If the portal or portal template folders in the target Content Server already contain content, the content is deleted before the import process begins.

> **Note:** the default behavior can be overwritten by specifying the `CheckExistingContents` property (values are `true` or `false`) in the properties file when running the Document Migration Utility specifying all the properties in a file. When set to `false`, the default behavior is overridden such that the target folder is not checked for any existing content and any existing content is not deleted. Care should be taken with having this setting as `false`; if there is any content in the target with the same Content ID or any content in the same folder with the same filename, the import will fail.

The XML document (`ExportImportData.xml`) located inside each top level folder is used to drive the import. This document describes the hierarchy of files and folders to recreate in the target Content Server. It also contains the metadata to be used when creating the artifacts and for documents it contains the path to the native file in the archive to use in the upload. For more information about the format of the archive, see Section F.2, "Understanding the Document Migration Utility."

On import, the following happens:

- The Document Migration Archive is unzipped into a temporary directory
- For each folder at the root of the archive (which represent a portal or portal template):
  - The corresponding folder is obtained from the target Content Server instance (using the folder name that is the GUID of the folder in Content Server)
  - The `ExportImportData.xml` document is read
  - For each folder or file described in the `ExportImportData.xml` the contents are recreated in the target instance

    The attributes in the `ExportImportData.xml` document for the folders and files are used as metadata for the contents being created and the files are uploaded from the corresponding location in the archive.
- At the end of a successful import, this temporary directory is deleted. If the import fails before completion, this temporary directory will remain.

F.2.2.1 Understanding How the Document Migration Utility Handles Metadata

The Document Migration Utility will maintain as much metadata as possible between the source and target. Most internal metadata (such as IDs), however, are not maintained. The following metadata is not included in the export set:

- Any date metadata, such as the creation date (for example, `dCreateDate`, `dReleaseDate`, `dDocLastModifiedDate`)
- For a document:
  - No internal identifiers for the document revisions or renditions (for example, `dID`, dDoc). Note that the content ID, `dDocName`, is included
  - The ID of the folder in which the document resides (for example, `xCollectionID`)
  - No Web-viewable renditions, as these will be recreated when the document is checked into the target
  - No information about the actual file (for example, its file type or mimetype) as these will get set when document is checked into the target
  - No status information, such as checked out, released state as these are irrelevant when creating the document in the target (for example, `dIsCheckedOut`, `dReleaseState`)
- For a folder:
  - No internal identifiers or paths such as the parent folder ID, or its own path (for example, `hasCollectionID`, `dCollectionID`, `dCollectionGUID`, `hasCollectionPath`, `dCollectionPath`)
- Storage rule, as this is dependent on the configuration of the target Content Server

- Security group or other security permissions, as these will get set when the content is created in the target (for example, `dSecurityGroup`, `dRead`, `canReadCollection`)

Metadata that is maintained:

- Names of artifacts, such as folder name or document title (for example, `dCollectionName`, `dDocTitle`)
- User name of person who created the content and modified it (for example, `dCollectionOwner`, `dDocOwner`)
- Wiki metadata, such as `dDocFunction`, `dDocType`, `xWCPageID`
- For a document:
  - The name of the file to upload for the document (`dOriginalName`)
  - The content ID (`dDocName`)

    If at the time of import a content item already exists in the target Content Server with the same content ID, import will fail. When rerunning an import with the property `CheckExistingContents=false` set, ensure that the target folder does not already contain the contents from the archive being imported to avoid a content ID error.
- Custom metadata (for example, `myCustomInteger`).

F.2.2.2 Document Migration Archive

The Document Migration Utility creates a single archive when exporting content from a source Content Server. The archive is used when running the Document Migration Utility to import the data into the target environment. The Document Migration archive has a specific layout and if an archive does not match this layout, the Document Migration Utility import will fail.

Archive format

The following describes the layout of the Document Migration Archive.

At the top level of the archive is a folder for each portal/portal template that has had their data exported.

- If a single portal/portal template has been exported there will only be one root folder, if multiple portals/portal templates have been exported there will be multiple root folders.
- The root folder names are the same as the internal GUID of the corresponding folder in Content Server (for example, `13828FE1-75D8-4A0B-A53B-A76AE78C1AE6`)

In each top level folder is an `ExportImportData.xml` document that describes the hierarchy of files and folders to recreate in the target Content Server for that portal or portal template. This document contains the metadata to be used when creating the artifacts (internal metadata, such as `xCollectionID`, `isreadonly`, `xStorageRule`, will not be included). For documents, it contains the path to the native file in the archive to use in the upload.

In each top-level portal and portal template folder are the exported documents. The layout of the documents inside the top-level folder mimics the layout in the source portal and portal template in the source Content Server. For example, if document `doc1.doc` resides in `folder1`, the top-level folder in the archive will contain a folder `folder1` in which `doc1.doc` resides. The documents file/folder structure in the

top-level folder must match the hierarchy described in the `ExportImportData.xml` so that the import code can find the native file for the documents described in that document uploading the document into the target. Empty folders in the portal or portal template folder in the source will not be in the archive. However, these folders are still described in the `ExportImportData.xml` document in order for them to be recreated in the target.

In a simplified overview, this would look like:

- Portal1 top level folder:
  - `ExportImportData.xml`
  - <Portal1's documents arranged in the same folder hierarchy as in the source>
- Portal2 top level folder:
  - `ExportImportData.xml`
  - <Portal2's documents arranged in the same folder hierarchy as in the source>

Understanding the ExportImportData.xml Document

The XSD for the `ExportImportData.xml` document is shown below.

XSD for `ExportImportData.xml`:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

<xs:element name="groupspace-folder" type="FolderType" />

  <!-- 'folders' must contain 1 or more 'folder' child elements -->
  <xs:complexType name="FoldersType">
    <xs:sequence>
      <xs:element name="folder" type="FolderType"
                  minOccurs="1" maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>

  <!-- 'documents' must contain 1 or more 'document' elements -->
  <xs:complexType name="DocumentsType">
    <xs:sequence>
      <xs:element name="document" type="DocumentType"
                  minOccurs="1" maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>

  <!-- 'attributes' must have 1 or more 'attribute' child elements -->
  <xs:complexType name="AttributesType">
    <xs:sequence>
      <xs:element name="attribute" type="AttributeType"
                  minOccurs="1" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- 'folder' has to have 1 and only 1 'attributes' child element
       0 or 1 'folders' child element, 0 or 1 'documents' child element -->
  <xs:complexType name="FolderType">
    <xs:sequence>
      <xs:element name="attributes" type="AttributesType"
                  minOccurs="1" maxOccurs="1" />
      <xs:element name="folders"    type="FoldersType"
```

```
                minOccurs="0" maxOccurs="1" />
      <xs:element name="documents"  type="DocumentsType"
                minOccurs="0" maxOccurs="1" />
    </xs:sequence>
  </xs:complexType>

  <!-- 'document' has to have : 1 and only 1 'attributes' child element
       and nothing else -->
  <xs:complexType name="DocumentType">
    <xs:sequence>
      <xs:element name="attributes" type="AttributesType"
                minOccurs="1" maxOccurs="1" />
    </xs:sequence>
  </xs:complexType>

  <!-- 'attribute' element has to have a 'name' and 'value' attributes -->
  <xs:complexType name="AttributeType">
    <xs:attribute name="name"  type="xs:string" use="required" />
    <xs:attribute name="value" type="xs:string" use="required" />
  </xs:complexType>

</xs:schema>
```

Where:

- **<groupspace-folder>** is the root tag that represents the portal or portal template folder.

  This tag contains the <attributes> tag, which in turn contains a number of attributes about the root folder and export data. These attributes are for information purposes only; they are *not* used in the import.

- **<attributes>** is used to group all the attributes of the document or folder.

  This tag must contain one or more <attribute> tags. No other child tags are permitted.

- **<attribute>** contains the metadata for a folder or document.

  This tag has two attributes:

  – name - the Content Server metadata name

  – value - the value of the metadata

  No child tags are permitted.

- **<folders>** is used to group all the folders in the current folder

  This tag must contain 1 or more <folder> tags. No other child tags are permitted.

- **<folder>** is used to indicate a child folder.

  This tag must have the <attributes> tag. If the folder has child folders, it will have the <folders> tag. If the folder has child documents, it will have the <documents> tag.

- **<documents>** is used to group all the documents in the current folder.

  This tag must contain one or more <document> tags. No other child tags are permitted.

- **<document>** is used to indicate a document in the current folder.

  This tag must have the <attributes> tag. No other child tags are permitted.

Annotated example:

The following annotated example shows a partially complete `ExportImportData.xml` document. Note that the example contains blank lines and XML comments that should not exist in a real `ExportImportData.xml` document.

```
<?xml version="1.0"encoding="UTF-8"standalone="yes"?>
<groupspace-folder>

  <attributes>
    <!-- Contains a set of attributes of the main portal folder -->
    <attribute name="export-date" value="2011-07-22 13:02:29"/>
  </attributes>

  <folders> <!-- only present if the portal contains any child folders -->
    <folder> <!-- a 'folder' tag exists for each child folder -->
      <attributes>
        <!-- contains the set of folder attributes, examples below -->
        <attribute value="F1" name="dCollectionName"/>
      </attributes>
      <!-- a 'folder' tag will contain the 'folders' tag if this folder
           contains child folders, i.e. if 'F1' has child folders -->
      <folders>
        <folder>
          <!-- attribute tags, child folders, child documents etc -->
        </folder>
      </folders> <!-- closing tag for all the folders in the
                      current folder à
      <!-- a 'folder' tag will contain the 'documents' tag if this folder
           contains documents, i.e. if 'F1' has documents at its root -->
      <documents>
        <document>
          <!-- attributes tags, see below -->
        </document>
      </documents> <!-- closing tag for all the documents in the
                        current folder -->
    </folder> <!-- closing tag for folder 'F1' -->
  </folders> <!-- closing tag for all the folders in the portal root -->

  <documents> <!- only present if the folder contains any documents
                 in the root folder -->
    <document> <!-- a 'document' tag exists for each document in the
                    folder -->
      <attributes>
        <!- contains the set of document attributes, examples below -->
        <attribute name="dDocTitle" value="Doc1"/>
        <attribute name="xForceFolderSecurity" value="TRUE"/>
        <attribute name="xSecurityClearanceLevel" value="public"/>
      </attributes>
    <document> <!-- closing tag for document 'Doc1' -->
  </documents><!-- closing tag for all the documents in the portal root -->

 </groupspace-folder>
```

Building the ExportImportData.xml Document

The following shows pseudo code for how the XML document is built:

```
- query Content Server for the portal's folder information
  - create a 'group-space-folder' node
    - create an 'attributes' node
    - for each of the three attributes we want to maintain (folder name, path
```

```
      and GUID)
        - create an 'attribute' node with name and value
        - add that 'attribute' node to the 'attributes' node
    - add that 'attributes' node to the 'group-space-folder' node

- query Content Server for the contents of the portal root folder
    - * If there are child folders
        - Create a 'folders' node
        - For each child folder
            - create a 'folder' node
            - create an 'attributes' node
            - for each folder metadata we want to maintain
                - create an 'attribute' node with name and value
                - add that 'attribute' node to the 'attributes' node
            - add the 'attributes' node to the 'folder' node
            - query Content Server for the contents of this current folder
                - go to * and repeat traversing folders which reside in this
                  current folder
                - go to # and repeat for documents which reside in this current
                  folder
             - add the 'folder' node to the 'folders' node
        - add the 'folders' node to the parent node (which will either be a
          'folder' node or the 'group-space-folder' node)

    - # If there are documents in this current folder
        - create a 'documents' node
        - for each document
            - create a 'document' node
            - create an 'attributes' node
            - for each metadata we want to maintain
                - create an 'attribute' node with name and value
                - add that 'attribute' node to the 'attributes' node
            - add the 'attributes' node to the 'document' node
            - add that 'document' node to the 'documents' node
         - add the 'documents' node to the 'folder' node

  - use jaxb to write the whole data to an xml file
```

Document Migration Archive Example

Below is an example of the archive format for a set of sample data and a sample `ExportImportData.xml` document for one of the portals in the example.

WebCenter Data

In the example below, two portals (Marketing and Sales) have been created in WebCenter Portal. In the Marketing portal two wiki pages have been created: `MarketingWiki` and `Tradeshows`. The Marketing portal has also had two folders created (Branding and Presentations), the latter of which contains subfolders with a PowerPoint document and a wiki document. The Sales portal has no wiki pages, but does contain three folders with some contents.

- Marketing
  Marketing portal contains the Content Server folder `GUID=29A4E019-7AE7-46A1-823B-AF16A313BBEF`
  - MarketingWiki
    * `MarketingWiki.htm` with the following metadata:

      ```
      dOriginalName="MarketingWiki.htm"
      dDocTitle="MarketingWiki.htm"
      ```

```
dDocFunction="wiki"
dDocType="Application"
xWCPageID="Page2.jspx"
```

  - Presentations
    - Branding

      `ProductBranding.pptx`

    - `Presentation Dates.htm` with the following metadata:
      (Note that here the wiki document is created inside a folder in a portal, rather than being created when a wiki page is created as for `MarketingWiki.htm` above)

      ```
      dOriginalName=" Presentation Dates.htm"
      dDocTitle="Presentation Dates"
      dDocFunction="wiki"
      dDocType="Application"
      xWCPageID = not set (as the wiki was not created when
      creating a wiki page)
      ```

    - `ProjectedDesigns.pptx`
  - Products
  - TradeShows
    - `TradeShows.htm` with the following metadata:

      ```
      dOriginalName=" TradeShows.htm"
      dDocTitle=" TradeShows.htm"
      dDocFunction="wiki"
      dDocType="Application"
      xWCPageID="Page4.jspx"
      ```

- Sales

  WebCenter Content folder `GUID = 629BFEBD-4E83-4DCA-895A-C72E5192FBFD`

  - DecConference
    - `GuestSpeakers.doc`
  - SalesConference
    - `Attendees.doc`
    - `TalkSchedules.doc`
  - `BudgetForcasts.doc`

Archive Contents

- `29A4E019-7AE7-46A1-823B-AF16A313BBEF` (contains Marketing portal contents)
  - MarketingWiki
    - `MarketingWiki.htm`
  - Presentations
    - Branding (`ProductBranding.pptx`)
    - `ProjectedDesigns.pptx`

        * Presentation Dates.htm
    - Tradeshows
        * TradeShows.htm
    - ExportImportData.xml
    - 2001Plans.doc
- 629BFEBD-4E83-4DCA-895A-C72E5192FBFD (contains Sales portal contents)
    - DecConference
        * GuestSpeakers.doc
    - SalesConference
        * Attendees.doc
        * TalkSchedules.doc
    - ExportImportData.xml
    - BudgetForcasts.doc

Note: There is no Products folder under the Marketing portal's root folder as this folder did not contain any documents.

ExportImportData.xml for the Marketing portal

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<groupspace-folder>
  <attributes>
    <attribute value="2011-10-27 15:50:18" name="export-date" />
    <attribute value="/WebCenter0711/Marketing/" name="group-space-path" />
    <attribute value="Marketing" name="group-space-name" />
    <attribute value="29A4E019-7AE7-46A1-823B-AF16A313BBEF"
               name="group-space-guid" />
  </attributes>
  <folders>
    <folder>
      <attributes>
        <attribute value="TRUE" name="xForceFolderSecurity" />
        <attribute value="sysadmin" name="dCollectionCreator" />
        <attribute value="MarketingWiki" name="dCollectionName" />
        <attribute value="0" name="isLink" />
        <attribute value="Page2.jspx" name="xWCPageId" />
        <attribute value="sysadmin" name="dCollectionOwner" />
        <attribute value="sysadmin" name="dCollectionModifier" />
        <attribute value="1" name="dCollectionEnabled" />
      </attributes>
      <documents>
        <document>
          <attributes>
            <attribute value="TRUE" name="xForceFolderSecurity" />
            <attribute value="Page2.jspx" name="xWCPageId" />
            <attribute value="wiki" name="dDocFunction" />
            <attribute value="Application" name="dDocType" />
            <attribute value="0" name="ishidden" />
            <attribute value="sysadmin" name="dDocOwner" />
            <attribute value="MarketingWiki.htm" name="dOriginalName" />
            <attribute value="MarketingWiki.htm" name="dDocTitle" />
```

```
        <attribute value="FALSE" name="xInhibitUpdate" />
        <attribute value="0" name="isreadonly" />
        <attribute value="0" name="CustomInteger" />
        <attribute value="sysadmin" name="dDocCreator" />
        <attribute value="sysadmin" name="dDocAuthor" />
        <attribute value="OWCSVR01USORAC012754" name="dDocName" />
        <attribute value="1" name="dRevisionID" />
        <attribute value="sysadmin" name="dDocLastModifier" />
      </attributes>
    </document>
  </documents>
</folder>
<folder>
  <attributes>
    <attribute value="TRUE" name="xForceFolderSecurity" />
    <attribute value="sysadmin" name="dCollectionCreator" />
    <attribute value="Presentations" name="dCollectionName" />
    <attribute value="0" name="isLink" />
    <attribute value="sysadmin" name="dCollectionOwner" />
    <attribute value="sysadmin" name="dCollectionModifier" />
    <attribute value="1" name="dCollectionEnabled" />
  </attributes>
  <folders>
    <folder>
      <attributes>
        <attribute value="TRUE" name="xForceFolderSecurity" />
        <attribute value="sysadmin" name="dCollectionCreator" />
        <attribute value="Branding" name="dCollectionName" />
        <attribute value="0" name="isLink" />
        <attribute value="sysadmin" name="dCollectionOwner" />
        <attribute value="sysadmin" name="dCollectionModifier" />
        <attribute value="1" name="dCollectionEnabled" />
      </attributes>
      <documents>
        <document>
          <attributes>
            <attribute value="TRUE" name="xForceFolderSecurity" />
            <attribute value="Document" name="dDocType" />
            <attribute value="sysadmin" name="dDocOwner" />
            <attribute value="0" name="ishidden" />
            <attribute value="ProductBranding.pptx" name="dDocTitle" />
            <attribute value="ProductBranding.pptx"
                       name="dOriginalName" />
            <attribute value="FALSE" name="xInhibitUpdate" />
            <attribute value="0" name="isreadonly" />
            <attribute value="0" name="CustomInteger" />
            <attribute value="sysadmin" name="dDocCreator" />
            <attribute value="sysadmin" name="dDocAuthor" />
            <attribute value="OWCSVR01USORAC012748" name="dDocName" />
            <attribute value="1" name="dRevisionID" />
            <attribute value="sysadmin" name="dDocLastModifier" />
          </attributes>
        </document>
      </documents>
    </folder>
  </folders>
  <documents>
      <document>
        <attributes>
          <attribute value="TRUE" name="xForceFolderSecurity" />
```

```
            <attribute value="wiki" name="dDocFunction" />
            <attribute value="Application" name="dDocType" />
            <attribute value="0" name="ishidden" />
            <attribute value="sysadmin" name="dDocOwner" />
            <attribute value="Presentation Dates" name="dDocTitle" />
            <attribute value="Presentation Dates.htm"
                       name="dOriginalName" />
            <attribute value="FALSE" name="xInhibitUpdate" />
            <attribute value="0" name="isreadonly" />
            <attribute value="0" name="CustomInteger" />
            <attribute value="sysadmin" name="dDocCreator" />
            <attribute value="sysadmin" name="dDocAuthor" />
            <attribute value="OWCSVR01USORAC012758" name="dDocName" />
            <attribute value="1" name="dRevisionID" />
            <attribute value="sysadmin" name="dDocLastModifier" />
          </attributes>
        </document>
        <document>
          <attributes>
            <attribute value="TRUE" name="xForceFolderSecurity" />
            <attribute value="Document" name="dDocType" />
            <attribute value="sysadmin" name="dDocOwner" />
            <attribute value="0" name="ishidden" />
            <attribute value="ProjectedDesigns.pptx" name="dDocTitle" />
            <attribute value="ProjectedDesigns.pptx"
                       name="dOriginalName" />
            <attribute value="FALSE" name="xInhibitUpdate" />
            <attribute value="0" name="isreadonly" />
            <attribute value="0" name="CustomInteger" />
            <attribute value="sysadmin" name="dDocCreator" />
            <attribute value="sysadmin" name="dDocAuthor" />
            <attribute value="OWCSVR01USORAC012747" name="dDocName" />
            <attribute value="1" name="dRevisionID" />
            <attribute value="sysadmin" name="dDocLastModifier" />
          </attributes>
        </document>
      </documents>
    </folder>
    <folder>
      <attributes>
        <attribute value="TRUE" name="xForceFolderSecurity" />
        <attribute value="sysadmin" name="dCollectionCreator" />
        <attribute value="Products" name="dCollectionName" />
        <attribute value="0" name="isLink" />
        <attribute value="sysadmin" name="dCollectionOwner" />
        <attribute value="sysadmin" name="dCollectionModifier" />
        <attribute value="1" name="dCollectionEnabled" />
      </attributes>
    </folder>
    <folder>
      <attributes>
        <attribute value="TRUE" name="xForceFolderSecurity" />
        <attribute value="sysadmin" name="dCollectionCreator" />
        <attribute value="TradeShows" name="dCollectionName" />
        <attribute value="0" name="isLink" />
        <attribute value="Page4.jspx" name="xWCPageId" />
        <attribute value="sysadmin" name="dCollectionOwner" />
        <attribute value="sysadmin" name="dCollectionModifier" />
        <attribute value="1" name="dCollectionEnabled" />
      </attributes>
```

```
        <documents>
          <document>
            <attributes>
              <attribute value="TRUE" name="xForceFolderSecurity" />
              <attribute value="Page4.jspx" name="xWCPageId" />
              <attribute value="wiki" name="dDocFunction" />
              <attribute value="Application" name="dDocType" />
              <attribute value="0" name="ishidden" />
              <attribute value="sysadmin" name="dDocOwner" />
              <attribute value="TradeShows.htm" name="dOriginalName" />
              <attribute value="TradeShows.htm" name="dDocTitle" />
              <attribute value="FALSE" name="xInhibitUpdate" />
              <attribute value="0" name="isreadonly" />
              <attribute value="0" name="CustomInteger" />
              <attribute value="sysadmin" name="dDocCreator" />
              <attribute value="sysadmin" name="dDocAuthor" />
              <attribute value="OWCSVR01USORAC012756" name="dDocName" />
              <attribute value="1" name="dRevisionID" />
              <attribute value="sysadmin" name="dDocLastModifier" />
            </attributes>
          </document>
        </documents>
      </folder>
    </folders>
  <documents>
    <document>
      <attributes>
        <attribute value="TRUE" name="xForceFolderSecurity" />
        <attribute value="Document" name="dDocType" />
        <attribute value="sysadmin" name="dDocOwner" />
        <attribute value="0" name="ishidden" />
        <attribute value="2011Plans.doc" name="dDocTitle" />
        <attribute value="2011Plans.doc" name="dOriginalName" />
        <attribute value="FALSE" name="xInhibitUpdate" />
        <attribute value="0" name="isreadonly" />
        <attribute value="0" name="CustomInteger" />
        <attribute value="sysadmin" name="dDocCreator" />
        <attribute value="sysadmin" name="dDocAuthor" />
        <attribute value="OWCSVR01USORAC012746" name="dDocName" />
        <attribute value="1" name="dRevisionID" />
        <attribute value="sysadmin" name="dDocLastModifier" />
      </attributes>
    </document>
  </documents>
</groupspace-folder>
```

F.3 Migrating Data from the Source Wiki Application to WebCenter Portal

To migrate content from an existing wiki application to WebCenter Portal, perform the following steps:

1. Prepare WebCenter Portal for import of the wiki content.
2. Write and run a 'Custom Wiki Extraction Tool' to extract content from the Wiki application into an archive matching the precise format expected by the Document Migration Utility.
3. Use the Document Migration Utility to import the archive into Content Server.
4. Create any wiki pages in WebCenter Portal to tie up with the content in Content Server.

These steps are described in more detail in the following topics:

- Section F.3.1, "Preparing WebCenter Portal for Importing Wiki Content"
- Section F.3.2, "Writing and Running a Custom Wiki Extraction Tool to Extract Content from the Wiki Application"
- Section F.3.3, "Using the Document Migration Utility to Import the Archive into the Target Portal"
- Section F.3.4, "Creating Wiki Pages in WebCenter Portal for the Content in Content Server"

F.3.1 Preparing WebCenter Portal for Importing Wiki Content

When the documents tool is enabled in a portal or portal template, a folder is created in Content Server for that portal or portal template. The GUIDs of these folders must be determined in order to construct the archive to be used with the Document Migration Utility. The folder GUIDs can be determined by following steps below:

1. Decide if you want to import all the wiki content into a single portal or multiple portals.
2. Log into WebCenter Portal and create the portals, taking note of the internal name of the portals.

   Ensure you are using a template that includes documents tool, otherwise you will have to enable the documents tool and setup the role permissions after portal creation.
3. Log into Content Server.
4. Ensure that the user's layout is **Top Menus**:
   a. Click the user's name to display the user's Profile page.
   b. Under **User Personalization Settings** check that **Layout** is set to `Top Menu`.
5. For each portal in which wiki content is to be imported, determine the folder GUID:
   a. Click **Browse Content**.
   b. Click on the root folder for the WebCenter Portal instance.

      This is the same as the **Root Folder** setting in the Content Server connection.
   c. Click the folder for the portal.

      The folder name will be the same as the portal's internal name.
   d. Click **Info** on the toolbar to display the folder information.
   e. Add `IsSoap=1` to the URL.
   f. Search for the string `dCollectionGUID`. For example:

      ```
      <idc:field
      name="dCollectionGUID">05573322-E895-EDA3-8A83-07CF39CBDE0
      5</idc:field>
      ```
6. Keep a note of the portal folder name and its GUID as the GUID is needed when building the archive in the next step.

F.3.2 Writing and Running a Custom Wiki Extraction Tool to Extract Content from the Wiki Application

To extract content from the source wiki application into an archive suitable for use with the Document Migration Utility, you'll need to write a custom application. For information about the format of the archive, see Section F.2, "Understanding the Document Migration Utility."

The custom wiki extraction tool must perform the following steps:

1. Extract and arrange the wiki content.

   Create a temporary directory and extract the wiki content from the source wiki application into it and arrange in the file system as it is to appear in WebCenter Portal.

2. Clean up the source HTML of wiki documents.

   For each wiki document, edit the HTML to remove application-specific HTML tags.

3. Re-write the URLs.

   For each wiki document, replace the existing URLs to content in the source wiki application to the URLs of the same artifacts that will be imported into WebCenter Portal.

4. Build the `ExportImportData.xml` documents.

   For each root folder build the `ExportImportData.xml` document which describes the data in the export set and is used to drive the import

5. Build the archive file.

   Create an archive of the manipulated wiki content that can be used to import the wiki content into WebCenter Portal.

Each of these steps is described more fully in the following topics:

- Section F.3.2.1, "Extracting and Arranging the Wiki Content"
- Section F.3.2.2, "Cleaning Up the Source HTML of Wiki Documents"
- Section F.3.2.3, "Rewriting the URLs"
- Section F.3.2.4, "Building the ExportImportData.xml Documents"
- Section F.3.2.5, "Building the Archive File"

F.3.2.1 Extracting and Arranging the Wiki Content

The wiki documents in the source application need to be extracted into a temporary directory on the file system and then arranged such that the file system mimics how the content is to be laid out in the target WebCenter Portal instance. If all the wiki documents are to be imported into a single portal, all of the content should be laid out under a single root folder named with the GUID of the corresponding portal folder in Content Server. If the wiki documents are to be imported into multiple portals, the content should be laid out under multiple root folders, each named with the GUID of their corresponding folder in Content Server. For more information on determining the GUID of a portal folder in Content Server, see Section F.3.1, "Preparing WebCenter Portal for Importing Wiki Content."

Note that when arranging the wiki content on the file system, you should consider how that content will be used in WebCenter Portal. For example:

- If wiki pages are to be created, then the wiki document for that wiki page must be located under a folder of the same name. For more information about wiki pages, see Section F.1, "Understanding Wiki Documents and Wiki Pages."
- When a folder contains a large number of contents, the rendering of that folder's contents could be impaired.
- Content Server has two settings that limit the number of folders and the number of files which can reside in a folder (**Maximum Folders Per Virtual Folder** and **Maximum Content Per Virtual Folder)**. When arranging your wiki content, ensure that a folder does not contain more folders than the folder limit setting or more documents than the document limit setting.

To create extracted wiki content, perform the following tasks:

1. Create root folders for each portal into which you will be importing the wiki documents, name the folders based on the GUID of the corresponding portal folder in Content Server.
2. For wiki documents for which wiki pages will be created in WebCenter Portal after import:
   a. Create a wiki folder with the same name as the wiki document.
   b. Place the wiki document in this folder.
   c. Place any other documents in this folder, if required.
   d. If there are related images and/or documents, add them to this wiki folder as well.
3. For any other wiki documents, create the folder hierarchy that will contain the documents.

Example:

Portal S1's folder in Content Server has a GUID of `21SD15F13B8_141D_421B_AD0e_BC54B6F16893`. After import, the `MarketingWiki` and `Tradeshows` wiki pages will be created and it is expected these wiki pages will show the `MarketingWiki.htm` and `Tradeshows.htm` wiki documents.

The following shows the organized structure of the extracted wiki documents and artifacts:

```
21SD15F13B8_141D_421B_AD0e_BC54B6F16893 (Root portal folder)
    Home.htm (Wiki document)
    MarketingWiki (Folder)
        MarketingWiki.htm (Wiki document)
    Branding (Folder)
        Presentation Dates.htm (Wiki document)
        Presentations (Folder)
    ProductBranding.pptx (File)
        ProjectedDesigns.pptx (File)
    Tradeshows (Folder)
        TradeShows.htm (Wiki document)
    Images (Folder)
        Image.jpg (Image)
```

F.3.2.2 Cleaning Up the Source HTML of Wiki Documents

In WebCenter Portal, the wiki editor will remove any HTML tags when the wiki page is being edited. Therefore it is advisable to remove any such HTML tags in the wiki documents prior to importing them into WebCenter Portal to avoid any confusion of

tags being removed when editing a wiki document after import. The following tags can be safely removed:

```
<html>, </html>
<head>, </head>
<meta>, </meta>
<title>, </title>
<body>, </body>
<tbody>, </tbody>
<thead>, </thead>
<tfoot>, </tfoot>
<script>, </script>
<link>, </link>
```

F.3.2.3 Rewriting the URLs

Wiki pages in the source wiki application may contain URLs referencing artifacts in within the source wiki application, such as links for embedded images or to other wiki page or documents. These artifacts will be migrated to the target WebCenter Portal instance and these links will need to be updated to reference the new artifact locations in the target WebCenter Portal instance.

The following types of URLs in the extracted wiki pages need to be changed to reference the URLs of the same artifacts in WebCenter Portal:

- Links to other Wiki pages
- Links to embedded images
- Links to documents

Follow the steps below to rewrite the URLs in the wiki documents:

1. Define attributes for the target WebCenter Portal instance that will be used in the URL replacement in step 3.
   - `WC_BASE_URL`: WebCenter instance base URL

     Example: `WC_BASE_URL=https://webcenter.example.com`
   - `UCM_ID`: The name of the connection in WebCenter Portal to the Content Server

     Example: `UCM_ID=dev_ucm`
   - `SPACE_GUID`: The GUID of the portal in WebCenter Portal where the content resides

     Example:
     `SPACE_GUID=s21sd15f13b8_141d_421b_ad0e_bc54b6f16893`

     For more information about determining the GUID, see Section F.3.1, "Preparing WebCenter Portal for Importing Wiki Content."
2. For each content item, define the item attributes that will be used in the URL replacement in step 3.
   - `FILE_NAME`: File name of the content item

     Example: `FILE_NAME=Home.htm`
   - `FILE_ID`: Unique Content Server content ID

     Example: `MARKETINGPORTAL1001`

Note that the FILE_ID must be unique across the entire Content Server instance. A suggested value is the name of the portal which the wiki documents are going to be imported into (with no portal in the name) post-fixed with a unique number (in the example above, the portal name was `Marketing Portal`).

3. Rewrite the URLs using the defined attributes as shown below:

   Embedded images

   - New URL format:

     ```
     IMG_REPLACE=img alt="FILE_NAME"
     resourceid="UCM_ID#dDocName:FILE_ID"
     src="WC_BASE_URL/webcenter/content/conn/UCM_ID/uuid/dDocNa
     me%3aFILE_ID"
     ```

   - Example:

     – Source URL:

       ```
       <img alt="image.jpg" src="images/image.jpg">
       ```

     – WebCenter URL:

       ```
       <img alt="image.jpg" resourceid="dev-ucm#dDoc-
       Name:WSIMPORT4" src="http://webcenter.example.com/web-
       center/content/conn/dev-ucm/uuid/dDocName%3aWSIMPORT4">
       ```

   Wiki pages

   - New URL format:

     ```
     URL_REPLACE=WC_BASE_URL/webcenter/faces/owResource.jspx?z=
     oracle.webcenter.doclib%21SPACE_GUID%21UCM_ID%2523dDocName
     %253AFILE_ID%21%21FILE_NAME
     ```

   - Example:

     – Source URL:

       ```
       <a href="Home.htm">Home</a>
       ```

     – WebCenter URL:

       ```
       <a href="http://webcenter.example.com/webcen-
       ter/faces/owResource.jspx?z=oracle.webcen-
       ter.doclib%21sd15f13b8_141d_421b_ad0e_bc54b6f16893%21de
       v-ucm%2523dDocName%253AWSIMPORT25%21%21Home.htm>Home</a
       >
       ```

   Links to documents

   - New URL format:

     ```
     DOCUMENT_REPLACE=WC_BASE_URL/webcenter/content/conn/UCM_ID
     /uuid/dDocName%3aFILE_ID
     ```

   - Example:

     – Source URL:

       ```
       <a href="MarketingWiki/Presentations/ProductBrand-
       ing.pptx"> Download Product Branding Presentation</a>
       ```

     – WebCenter URL:

```
<a href="http://webcenter.example.com/webcenter/con-
tent/dev-ucm/uuid/dDocName%3aWSIMPORT7"> Download Prod-
uct Branding Presentation</a>
```

F.3.2.4 Building the ExportImportData.xml Documents

In each root folder containing the contents to be imported an `ExportImportData.xml` document needs to be created. The `ExportImportData.xml` document describes the contents of the root folder and is used to drive the import when importing the content into WebCenter Portal using the Document Migration Utility. For more information about the Document Migration Utility and the `ExportImportData.xml` document, see Section F.2, "Understanding the Document Migration Utility."

Any metadata to be created with the document on import must be specified in the `ExportImportData.xml` document. In WebCenter Portal, wiki documents are stored as HTML documents but have extra metadata to identify them as wiki documents rather than normal HTML documents. Ensure the `ExportImportData.xml` document has this metadata specified for all wiki documents in the extracted contents. For more information about the metadata required for wiki document, see Section F.1, "Understanding Wiki Documents and Wiki Pages."

Note: A content ID (`dDocName`) is automatically generated by Content Server when a document is checked in without one being specified. If you wish your documents to have fixed content IDs, include the `dDocName` metadata with the document metadata in the `ExportImportData.xml` document. The `dDocName` must be unique across the whole Content Server or document check in will fail. A suggestion is to chose your own prefix for the content ID and append numbers incrementally to the end.

The `ExportImportData.xml` document can be generated manually for each root folder. Alternatively, you can write a custom script to traverse through the root folder contents and generate the document.

It is imperative for the structure of the contents on the file system is detailed in `ExportImportData.xml` document correctly. If there is a mismatch between the hierarchy of contents described in the `ExportImportData.xml` document and the file system, the import into the portal folder in the target Content Server will fail.

Example:

In this example a custom script named `convert_program` traverses through a root folder called `21SD15F13B8_141D_421B_AD0e_BC54B6F16893` and creates an `ExportImportData.xml` document in the current working directory detailing the contents of the folder.

```
cd 21SD15F13B8_141D_421B_AD0e_BC54B6F16893
run convert_program
```

F.3.2.5 Building the Archive File

Create an archive of the extracted and manipulated wiki documents by zipping up the root portal folders. The zip archive must have the root folders inside the archive rather than just the contents of the root folders. One zip file can contain multiple root folders for different portals, or you can create one zip file for each root folder.

Example:

In the following example, wiki documents have been extracted and manipulated in a folder called `21SD15F13B8_141D_421B_AD0e_BC54B6F16893` in the folder `/scratch/wikiexports` and the archive to create is `wsimport.zip`.

```
cd /scratch/wikiexports
zip -r wcimport.zip 21SD15F13B8_141D_421B_AD0e_BC54B6F16893/
```

Note: Ensure that the archive does not exist prior to zipping up the folder contents as some zip tools will add content to the specified archive if it already exists rather than overwriting the archive.

F.3.3 Using the Document Migration Utility to Import the Archive into the Target Portal

Run the Document Migration Utility specifying the archive generated in the previous step to import the content into the target Content Server. For information about using the Document Migration Utility, see Section F.3.3.2, "Migrating Content Using the Document Migration Utility."

Log into WebCenter Portal and navigate to the portals to which content was imported and ensure the content exists.

F.3.3.1 Properties Required to Run the Document Migration Utility

Table F–1 describes the properties required to run the Document Migration Utility. For information on how to run the utility, see Section F.3.3.2, "Migrating Content Using the Document Migration Utility."

Table F–1 Document Migration Properties

Property	Description	Requirement
Usage	Specifies whether you want to import or export content to a file. Options are: `import` and `export`	Export and Import
MDSConn	Specifies MDS JDBC connection in the format: `jdbc:oracle:thin:@host:port:SID` or `jdbc:oracle:thin:@host:port/ServiceName`	Export
MDSUser	Specifies the MDS user name used by WebCenter Portal.	Export
MDSPwd	Specifies password for the MDS user. Only include to avoid password prompt.	Export
ExportScopes	Specifies the internal name of each portal/portal template with content to export. Separate multiple portal/template names with a comma. Prefix portal template names with `spacetemplate/<template_internal_name>`. Ensure there are no spaces in the comma separated list. You can obtain internal names from the *About Portal* and *About Portal Template* dialogs. Do not enter display names here.	Export

Table F–1 (Cont.) Document Migration Properties

Property	Description	Requirement
UCMConn	Specifies Content Server URL in the format: `idc://host:intradocPort` When `usage=export`, specify the URL of the Content Server instance from which content is to be exported. When `usage=import`, specify the URL of the Content Server instance to which the content is to be imported.	Export and Import
UCMUser	Specifies the Content Server user name used to connect through RIDC. This user must have sufficient privileges to perform the export or import; either a user defined in an external identity store or the Content Server administrator `sysadmin`.	Export and Import
UCMPwd	Specifies password for the Content Server user. Only include to avoid password prompt.	Export and Import
TmpDirPath	Optional. Temporary location for data extraction. If not specified, defaults to the system `tmp` directory.	Export and Import
ArchivePath	Document archive location.	Export and Import
ArchiveName	Optional. Name for the document archive (`.zip`). Default is `docsexport.zip`.	Export and Import

F.3.3.2 Migrating Content Using the Document Migration Utility

You can use any of the following methods:

- Specifying Document Migration Properties in a Properties File
- Specifying Document Migration Properties on the Command Line
- Specifying Document Migration Properties on the Command Line When Prompted

Specifying Document Migration Properties in a Properties File

1. Create a properties file containing all the properties required for your export/import. See Table F–1 for a description of all the properties.
   a. Copy and paste the following properties file into Notepad or another suitable text editor, then edit according to your environment:

```
# Document migration properties.

# Specify whether you want to export content to a file or
# import content from an archive to another content repository
# valid values: export | import
Usage=export

# Specify connection details for Oracle WebCenter Content repository:
#   UCMConn - Content Server URL. Format: idc://host:intradocPort
#   UCMUser - Content Server user name used to connect through RIDC
#   UCMPwd  - Password for UCMUser. Only include to avoid password prompt
# Required for: Export and Import

UCMConn=idc://mycontentserver.mycompany.com:9444
UCMUser=<enter Content Server admin user name here>
#UCMPwd=<enter password for UCMUser here>

# Specify a temp directory and name/location for the export archive
```

```
#   TmpDirPath  -Optional. Temporary location for data extraction.
#               If not specified, defaults to the system temporary
directory.
#   ArchiveName -Optional. Name for the document archive (.zip).
#               Default is docsexport.zip.
#   ArchivePath -Document archive location
# Required for: Export and Import

TmpDirPath=/scratch/user1/migrateMyPortalDocs/tmpdir
ArchivePath=/scratch/user1/migrateMyPortalDocs/output
ArchiveName=myportaldocs.zip

# Specify MDS details (export only)
#   MDSConn - MDS JDBC connection. Format:
#               jdbc:oracle:thin:@host:port:SID   or
#               jdbc:oracle:thin:@host:port/ServiceName
#   MDSUser - MDS schema user name used by the WebCenter Portal application
#   MDSPwd = Password for MDSUser. Only include to avoid password prompt
# Required for: Export

MDSConn=jdbc:oracle:thin:@mymdshost.mycompany.com:1521:wkcdb01
MDSUser=<enter MDS user name here>
#MDSPwd=<enter password for MDSUser here>

# Specify target portal for export or import.
# Seperate multiple portal/template names with a comma.
# Use internal names only. Do not enter display names.
# Obtain internal names from "About Portal" and "About Portal Template"
dialogs.
# Prefix portal template names with
'spacetemplate/<template_internal_name>'
# as indicated in the example.
# Required for: Export

ExportScopes=MyPortal1,MyPortal2,spacetemplate/MyPortalTemplate
```

b. Save the file. For example, save as `myMigrationProperties.properties` or similar.

2. Navigate to the *WCP_ORACLE_HOME*`/webcenter/archives` directory in which the Document Migration Utility, `content-migration-tool.jar` is located.
3. Run the Document Migration Utility by specifying the absolute path to your document migration properties file on the command line (Example F–1):

```
java -jar content-migration-tool.jar
<absolute_path_to_migrationPropertiesFilename>
```

Optionally, specify logging settings using the `java.util.logging.config.file` parameter as described in Section F.3.3.3, "Running the Document Migration Utility with Additional Logging."

Example F–1 Specifying Document Migration Properties in a Properties File

```
java -jar content-migration-tool.jar /home/user1/myMigrationProperties.properties
```

Specifying Document Migration Properties on the Command Line

1. Navigate to the *WCP_ORACLE_HOME*`/webcenter/archives` directory in which the Document Migration Utility, `content-migration-tool.jar` is located.

2. Run the Document Migration Utility by specifying individual properties on the command line:

   To export content:

   ```
   java -jar content-migration.jar Usage UCMConn UCMUser TmpDirPath ArchivePath
   ArchiveName MDSConn MDSUser ExportScopes [UCMPwd MDSPwd]
   ```

   To import content:

   ```
   java -jar content-migration.jar Usage UCMConn UCMUser TmpDirPath ArchvePath
   ArchiveName [UCMPwd]
   ```

   Note: You can, optionally, specify the `UCMPwd` and `MDSPwd` parameters on the command line. If you do not do so, you are prompted to provide them.

   Optionally, specify logging settings using the `java.util.logging.config.file` parameter, as described in Section F.3.3.3, "Running the Document Migration Utility with Additional Logging."

Specifying Document Migration Properties on the Command Line When Prompted

1. Navigate to the `WCP_ORACLE_HOME/webcenter/archives` directory in which the Document Migration Utility, `content-migration-tool.jar` is located.
2. Run the Document Migration Utility by specifying the properties on the command line when prompted:

   ```
   java -jar content-migration.jar
   ```

   Optionally, specify logging settings using the `java.util.logging.config.file` parameter, as described in Section F.3.3.3, "Running the Document Migration Utility with Additional Logging."

F.3.3.3 Running the Document Migration Utility with Additional Logging

You can optionally run the Document Migration Utility with additional logging using the `java.util.logging.config.file` parameter as follows:

```
java -Djava.util.logging.config.file=<absolute_path_to_logging_properties_file>
-jar content-migration-tool.jar <migrationProperties>
```

Note: The `java.util.logging.config.file` parameter must be specified immediately after the `java` command and before `-jar`.

Where the `logging_properties_file` includes settings such as:

```
handlers=java.util.logging.ConsoleHandler.level=INFO
java.util.logging.ConsoleHandler.level=FINER
java.util.logging.ConsoleHandler.formatter=java.util.logging.SimpleFormatter
oracle.webcenter.doclib.level=INFO
```

F.3.4 Creating Wiki Pages in WebCenter Portal for the Content in Content Server

To use WebCenter Portal wiki pages to display the imported wikis, perform the steps below. For more information about wiki pages, see Section F.1, "Understanding Wiki Documents and Wiki Pages."

1. Log into WebCenter Portal.
2. Locate the portal where the content has been uploaded.
3. Click **Actions** and select **Create and Page**.
4. Give the wiki page a **Name** and select the `Wiki` page layout.

Note that the name of the wiki page must match the name of the folder in the portal folder in WebCenter Portal, which contains the wiki page of the same name.

For example, if in the portal folder you have a `MarketingWiki` folder and a `MarketingWiki.htm` document, the name of the wiki page must be `MarketingWiki`.

G

Troubleshooting Oracle WebCenter Portal

This appendix presents troubleshooting information for Oracle WebCenter Portal.

This appendix contains the following topics:

G.1 Troubleshooting Roadmap

Use this documentation roadmap to find troubleshooting information for Oracle WebCenter Portal. In Figure G–1, find the starting point that best describes your issue, then click the graphic or use the links in Table G–1 to jump to the section that you need.

Figure G–1 Troubleshooting Oracle WebCenter Portal Roadmap

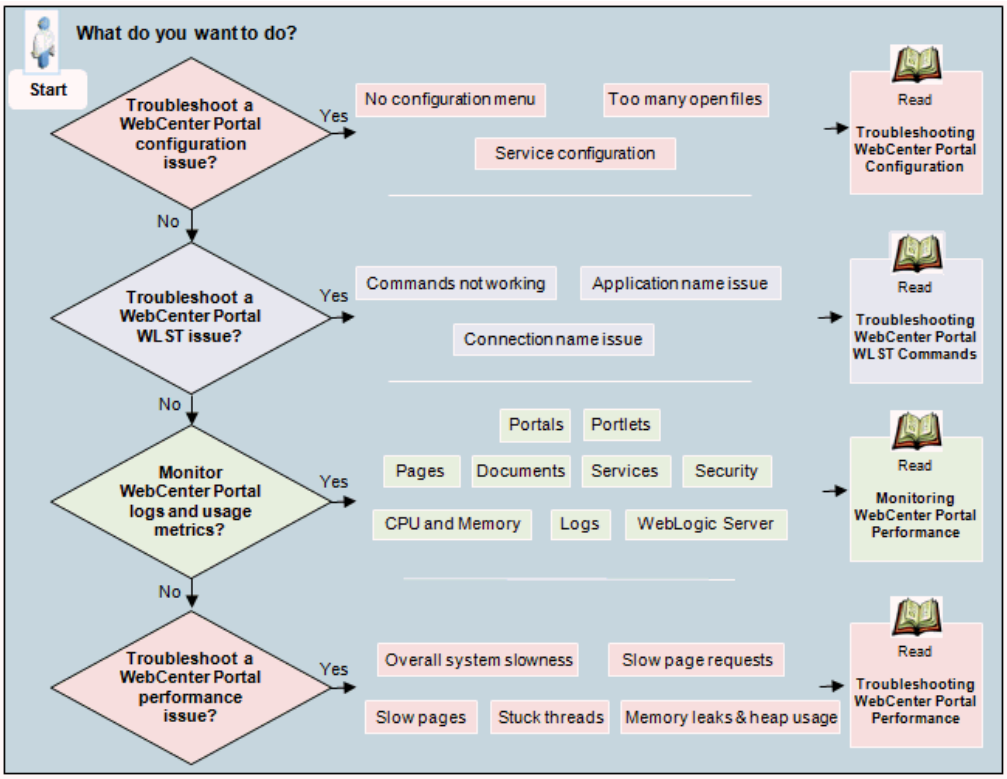


Table G–1 Starting Points for Troubleshooting WebCenter Portal

What do you want to do?	Link to Troubleshooting Section in the Guide
▪ Troubleshoot Oracle WebCenter Portal configuration issues?	Section G.2, "Troubleshooting Oracle WebCenter Portal Configuration Issues"
▪ Troubleshoot Oracle WebCenter Portal WLST issues?	Section G.3, "Troubleshooting Oracle WebCenter Portal WLST Command Issues"
▪ Monitor Oracle WebCenter Portal logs and metrics?	Section 27, "Monitoring Oracle WebCenter Portal Performance" Section 28, "Managing Oracle WebCenter Portal Logs"
▪ Troubleshoot Oracle WebCenter Portal performance issue?	Section G.4, "Troubleshooting Oracle WebCenter Portal Performance Issues"

G.2 Troubleshooting Oracle WebCenter Portal Configuration Issues

This section includes the following subsections:

- Section G.2.1, "How Do I Find Out Which Oracle WebCenter Portal Version Is Installed?"
- Section G.2.2, "WebCenter Portal Menu Does Not Display in Fusion Middleware Control"
- Section G.2.3, "Configuration Options Unavailable"
- Section G.2.4, "Configuration Issues with One or More Tools or Services"
- Section G.2.5, "Configuration for One Application Reflects in Another"
- Section G.2.6, "Logs Indicate Too Many Open Files"

G.2.1 How Do I Find Out Which Oracle WebCenter Portal Version Is Installed?

Always use Oracle's OPatch utility to obtain version information for Oracle WebCenter Portal products and components installed in your environment.

To run OPatch command with the `lsinventory` option:

1. Set Opatch environment variables *`WCP_ORACLE_HOME`*, *`MW_HOME`*, and *`PATH`*.

   See the "Patching Oracle Fusion Middleware with Oracle OPatch" section in *Oracle Fusion Middleware Patching Guide*.

2. Run the following OPatch command

   ```
   opatch lsinventory -details
   ```

   The output lists which components are installed and their version numbers, similar to that shown here:

   ```
   ...
   Oracle Upgrade Assistant for Webcenter            11.1.1.8.0
   Oracle WebCenter Portal Suite                     11.1.1.8.0
   Oracle WebCenter Portal Suite 11g                 11.1.1.8.0
   Oracle WebCenter Portal: Activity Graph           11.1.1.8.0
   Oracle WebCenter Portal: Analytics Collector      11.1.1.8.0
   Oracle WebCenter Portal: Discussions Server       11.1.1.8.0
   Oracle Webcenter Portal: Framework                11.1.1.8.0
   Oracle Webcenter Portal: Framework Core           11.1.1.8.0
   Oracle WebCenter Portal: Pagelet Producers        11.1.1.8.0
   Oracle WebCenter Portal: Personalization          11.1.1.8.0
   Oracle Webcenter Portal: Portlet Server           11.1.1.8.0
   Oracle WebCenter Portal: RCU                      11.1.1.8.0
   Oracle Webcenter Portal: Spaces                   11.1.1.8.0
   Oracle WebCenter Portal: Suite Components         11.1.1.8.0
   Oracle WebCenter Portal: Wiki                     11.1.1.8.0
   ...
   ```

   Note: `Oracle WebCenter Portal Suite 11g` is a child component of `Oracle WebCenter Portal Suite` and the versions are always the same.

See the "Patching Oracle Fusion Middleware with Oracle OPatch" section in *Oracle Fusion Middleware Patching Guide* and the "Lsinventory Command for OUI-based Oracle Homes" section in *Oracle Universal Installer and OPatch User's Guide for Windows and UNIX*.

Note: Always use OPatch to obtain the version number. Versions that display alongside Oracle WebCenter Portal product and component names in Fusion Middleware Control do not reflect the true version number of installed products, as shown in Figure G–2 and Figure G–3.

Figure G–2 Incorrect Version Number for WebCenter Portal in Fusion Middleware Control

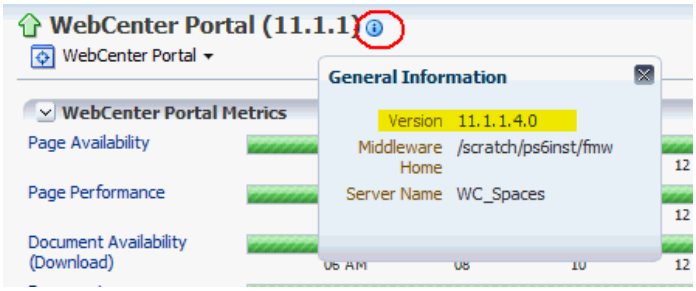
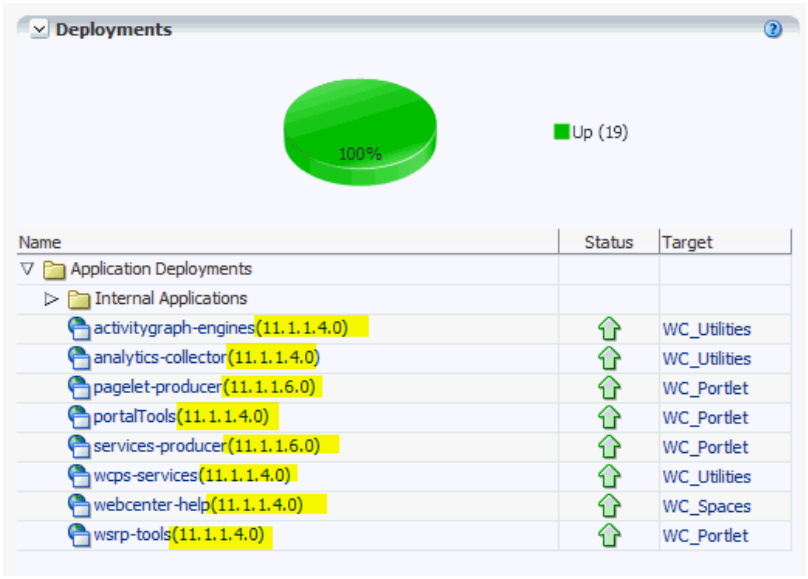


Figure G–3 Incorrect Version Numbers for Other Applications in Fusion Middleware Control



G.2.2 WebCenter Portal Menu Does Not Display in Fusion Middleware Control

Problem

After logging into Fusion Middleware Control, you cannot find the **WebCenter Portal** option in the **Application Deployment** menu for your application.

Solution

Ensure the following:

- Deployed application is a Portal Framework application, created using the *WebCenter Portal Framework Application* template in JDeveloper.

  The **WebCenter Portal** option only displays for applications developed using the *WebCenter Portal Framework Application* template in JDeveloper.

- Deployed Portal Framework application is up and running.

- Deployed Portal Framework application contains accurate information about the MDS repository and partition, and the MDS repository is accessible to the application. To verify this information, check the `metadata-store-usages` section in the `adf-config.xml` file. For information on MDS, see the

"Understanding the MDS Repository" section in *Oracle Fusion Middleware Administrator's Guide*.

- Portal Framework application is packaged with required artifacts to support configuration:
  - `adf-jndi-config` namespace is configured in the application's `adf-config.xml` file. This is provisioned at design time. The following is an example (the text in **bold**) of the `adf-jndi-config` namespace:

```
<adf-config xmlns="http://xmlns.oracle.com/adf/config"
     xmlns:jndiC="http://xmlns.oracle.com/adf/jndi/config"
     xmlns:ns2="http://xmlns.oracle.com/mds/config"
     xmlns:ns3="http://xmlns.oracle.com/adf/mds/config">
  ...
  ...
</adf-config>
```

  - Appropriate listeners exist in the `web.xml` file to register the MBeans. This is provisioned at design time. For example, see the text in **bold** in the following snippet of the `web.xml` file:

```
<listener>
   <description>ADF Config MBeans</description>
   <display-name>ADF Config MBeans</display-name>

<listener-class>oracle.adf.mbean.share.config.ADFConfigLifeCycleCallBack</listener-class>
</listener>
<listener>
    <description>ADF Connection MBeans</description>
    <display-name>ADF Connection MBeans</display-name>

<listener-class>oracle.adf.mbean.share.connection.ADFConnectionLifeCycleCallBack</listener-class>
</listener>
```

- `ADFConfig` and `ADFConnection` MBeans are registered for WebCenter Portal or your Portal Framework application. You can verify whether these MBeans are registered through the System MBean Browser:

  1. In Fusion Middleware Control, open the System MBean Browser for your application. Do one of the following:

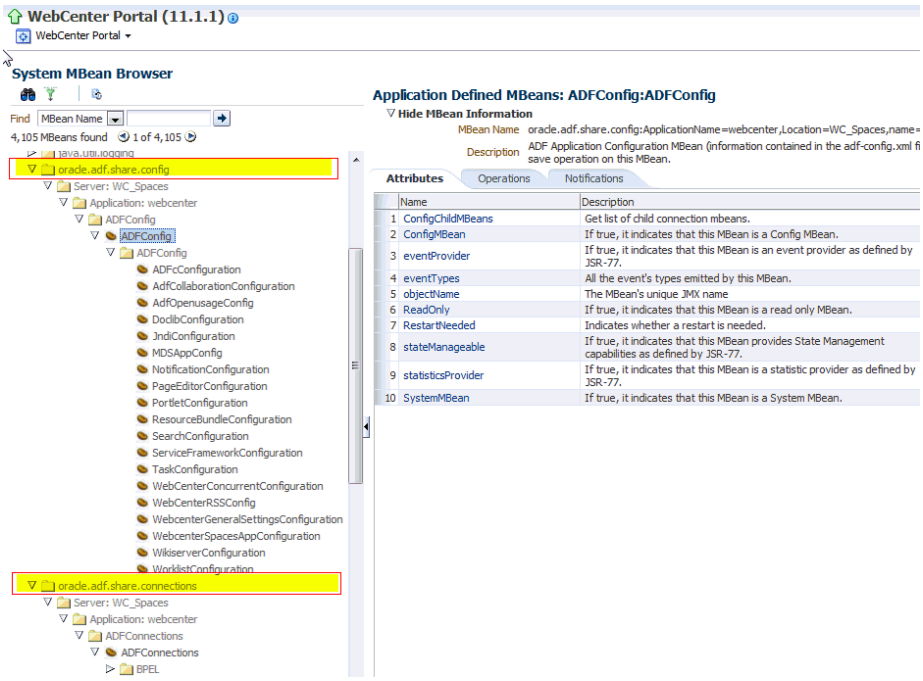
     For WebCenter Portal, select the menu option **WebCenter Portal** >**System MBean Browser**.

     For a Portal Framework application, select the menu option **Application Deployment** > **System MBean Browser**.

  2. Locate `ADFConnection` MBeans for your application under **Application Defined MBeans** > **oracle.adf.mbean.share.connection**, as shown in Figure G–4.

  3. Similarly, locate `ADFConfig` MBeans for your application under **Application Defined MBeans** > **oracle.adf.mbean.share.config**, as shown in Figure G–4.

Figure G–4 Application Defined MBeans



See also, Section 1.13.4, "System MBean Browser."

- Review the latest configuration in `adf-connections.xml` and `adf-config.xml` to check the content is correct. Some typical problems include:
  - Configuration file is not compliant with its XML schema For example, there are duplicate configuration elements when the schema only allows for 1 occurrence.
  - XML namespace is missing for configuration referenced within the file.
  - XML element is not qualified with the XML namespace.

  See also, Appendix A "Exporting Configuration Files with MDS Customizations".
- Check the application's diagnostic logs, analyze messages for the modules `oracle.adf.mbean.share.connection` and `oracle.adf.mbean.share.config`, and determine what must be done:
  - For WebCenter Portal, the log file is available in the *`DOMAIN_HOME`*`/servers/`*`ServerName`*`/logs` directory. The log file follows the naming convention of *`ServerName`*`-diagnostic.log`. See also, Section 28.2.1, "Viewing and Configuring WebCenter Portal Logs."
  - For Portal Framework applications, the log file is available in the *`DOMAIN_HOME`*`/servers/`*`ServerName`*`/logs` directory. The log file follows the naming convention *`ServerName`*`-diagnostic.log`. See also, Section 28.2.2, "Viewing and Configuring Portal Framework Application Logs."

G.2.3 Configuration Options Unavailable

Problem

When you try to configure WebCenter Portal or a Portal Framework application through Fusion Middleware Control, the following message displays:

```
Configuration options currently unavailable. The application application_name
might be down, did not start-up properly, or is incorrectly packaged.
Check the log files for further details.
```

For example, you try to change options available through the **Application Settings** screen or configure connections through the **WebCenter Portal Service Configuration** screen in Fusion Middleware Control.

Solution

Check the application's diagnostic logs:

- For WebCenter Portal, the log file is available in the *DOMAIN_HOME*/servers/*ServerName*/logs directory. The log file follows the naming convention of *ServerName*-diagnostic.log. See also, Section 28.2.1, "Viewing and Configuring WebCenter Portal Logs."
- For Portal Framework applications, the log file is available in the *DOMAIN_HOME*/servers/*ServerName*/logs directory. The log file follows the naming convention *ServerName*-diagnostic.log. See also, Section 28.2.2, "Viewing and Configuring Portal Framework Application Logs."

Analyze messages for the modules oracle.adf.mbean.share.connection and oracle.adf.mbean.share.config, and determine what must be done.

See also, Section G.2.2, "WebCenter Portal Menu Does Not Display in Fusion Middleware Control."

G.2.4 Configuration Issues with One or More Tools or Services

Do not attempt to configure services that your application does not support. WebCenter Portal configuration through Enterprise Manager and WLST fails if you try to configure a service, say discussions, that your application was not designed to use.

The out-of-the-box application, WebCenter Portal, is designed to support all tools and services but Portal Framework applications that you build using JDeveloper only provide artifacts in the application's configuration for services that the developer specifically included during design time. For example, if the developer did not add or configure discussions for the application through JDeveloper, you cannot configure discussions postdeployment through Enterprise Manager and WLST.

If you are having issues configuring or connecting to one or more tool or service, refer to the appropriate troubleshooting section:

- Section 9.13, "Troubleshooting Issues with Content Repositories"
- Section 10.8, "Troubleshooting Issues with Recommendations"
- Section 11.11, "Troubleshooting Issues with Analytics"
- Section 12.13, "Troubleshooting Issues with Announcements and Discussions"
- Section 13.9, "Troubleshooting Issues with Events"
- Section 15.11, "Troubleshooting Issues with Mail"
- Section 19.6, "Troubleshooting Issues with Notifications"

- Section 18.7, "Troubleshooting Issues with Oracle SES"
- Section 20.7, "Troubleshooting Issues with Worklists"
- Section 21.13, "Troubleshooting Portlet Producer Issues"
- Section 21.12.3, "Troubleshooting WebCenter Services Portlets"
- Section 30.3, "Troubleshooting Security Configuration Issues"
- Chapter G.7, "Troubleshooting WebCenter Portal Import and Export"
- Chapter G.8, "Troubleshooting Individual Portal and Portal Template Import and Export"

G.2.5 Configuration for One Application Reflects in Another

Problem

You configured WebCenter Portal or a Portal Framework application, but those configurations also show in another application.

For example, you created or edited a mail connection for a Portal Framework application named MyPortalApp1 and you discover that the connection changes are also seen in another application MyPortalApp2

Solution

This happens when multiple applications share the MDS partition in the same schema. To resolve this problem, deploy these applications again and ensure that each application either uses a different MDS schema or a different MDS partition. For information about creating a MDS repository or configuring an existing application to use a different MDS repository or partition, see the "Managing the Metadata Repository" section in *Oracle Fusion Middleware Administrator's Guide*.

G.2.6 Logs Indicate Too Many Open Files

Problem

WebCenter Portal or a Portal Framework application is inaccessible or displaying error messages and the diagnostic log files indicates that there is an issue with '`too many open files`'.

Solution

Do the following:

- Check the number of file handles configured on each of the back-end servers, primarily the database, and increase appropriately.
- If the problem persists after increasing the file handles, check the value of `fs.file-max` in the `/etc/sysctl.conf` file and increase the value appropriately.

G.3 Troubleshooting Oracle WebCenter Portal WLST Command Issues

This section includes the following subsections:

- Section G.3.1, "No Oracle WebCenter Portal WLST Commands Work"
- Section G.3.2, "WLST Commands Do Not Work for a Particular Tool or Service"

- Section G.3.3, "Connection Name Specified Already Exists"
- Section G.3.4, "WLST Shell is Not Connected to the WebLogic Server"
- Section G.3.5, "More Than One Application with the Same Name Exists in the Domain"
- Section G.3.6, "More Than One Application with the Same Name Exists on a Managed Server"
- Section G.3.7, "Already in Domain Runtime Tree Message Displays"

See also, Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

G.3.1 No Oracle WebCenter Portal WLST Commands Work

Problem

You are unable to run any WLST commands.

Solution

Ensure the following:

- Always run Oracle WebCenter Portal WLST commands from your **WebCenter Portal Oracle home directory** (*WCP_ORACLE_HOME*/common/bin).

  If you attempt to run Oracle WebCenter Portal WLST commands from the wrong directory you will see a `NameError`.

- No files other than Python are stored in the WLST source directory: *WCP_ORACLE_HOME*/common/bin/wlst. This directory must contain files with the `.py` extension only.

  The default set of files in this location contain legal Python files from Oracle. It is possible that a user copied some non-python script to this directory, for example, a backup file or a test python file with syntax errors.

- `webcenter-wlst.jar` is located at *WCP_ORACLE_HOME*/common/bin/wlst/lib.

See also, Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

G.3.2 WLST Commands Do Not Work for a Particular Tool or Service

Problem

You are unable to run WLST commands for a particular tool or service, and therefore, you cannot configure that tool/service.

Solution

First, run generic non-Oracle WebCenter Portal commands, for example, run `listApplications` or `displayMetricTableNames` to verify whether these commands work. If generic commands do not work, then apply the solution described in Section G.3.1, "No Oracle WebCenter Portal WLST Commands Work."

If generic commands work, then run test commands to check Oracle WebCenter Portal-specific commands for syntax errors. Run the appropriate WSLT check command (see Table G–2).

Table G–2 File Names and WLST Commands for Oracle WebCenter Portal Tools and Service

Service Name	File Name	WLST Command
Activity Graph	`ActivityGraph.py`	`metadataAdminCheck()`
Activity Stream	`ActivityStream.py`	`asCheck()`
Analytics	`Analytics.py` `OpenUsage.py`	`analyticsCheck()` `openusageCheck()`
Discussions and Announcements	`Forum.py` `JiveAdmin.py`	`fcpCheck()`
Documents	`Doclib.py`	`doclibCheck()`
External Applications	`ExtApp.py`	`extCheck()`
Portal Events	`Community.py`	`ceCheck()`
Instant Messaging and Presence	`Imp.py`	`rtcCheck()`
Mail	`Mail.py`	`mailCheck()`
Notifications	`Notification.py`	`notificationCheck()`
Personal Events	`Personal.py`	`peCheck()`
Producers		
PDK-Java Producers	`Pdk.py`	`pdkCheck()`
WSRP Producers	`Wsrp.py`	`wsrpCheck()`
Pagelet Producers	`Ensemble.py`	`ensembleCheck()`
Producer Helper	`Producer.py`	`producerHelperCheck()`
RSS News Feed	`RSS.py`	`rssCheck()`
Search	`Ses.py`	`sesCheck()`
Worklist	`Bpel.py`	`bpelCheck()`
Export/Import - WebCenter Portal applications	`Lifecycle.py`	`lifecycleCheck()`
Export/Import - Portals and Template	`ExtImp.py`	`expimpCheck()`
Synchronize Users	`SynchronizeUser.py`	`userRenameCheck()`
Rename Users	`UserRename.py`	`userRenameCheck()`
WebCenter Portal - General		
Service Framework	`WcServiceFwk.py`	`serviceFwkCheck()`
General Settings	`WebCenterGeneralSettings.py`	`generalSettingsCheck()`
WebCenter Portal and SOA	`WebCenterSpacesSOA.py`	`spaceCheck()`

See also, Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

For more information about Oracle WebCenter Portal's WLST commands, see the "WebCenter Portal Custom WLST Commands" chapter in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

G.3.3 Connection Name Specified Already Exists

Problem

You are unable to create a connection with the name *Connection_Name*. The following message displays:

```
A connection with name Connection_Name already exists.
```

For example, you try to create an external application connection using the WLST command `createExtAppConnection` or connect to a mail server using `createMailConnection`.

Solution

Connection names must be unique (across all connection types) within WebCenter Portal or a Portal Framework application. This error occurs when you try to create a connection with a name that is in use. Ensure that you use a unique name for your connection.

G.3.4 WLST Shell is Not Connected to the WebLogic Server

Problem

You must connect to the Administration Server for Oracle WebCenter Portal before running WLST commands. Oracle WebCenter Portal WLST commands do not work without a connection.

Solution

Run the following command to connect the WLST shell to the managed server:

```
connect(username, password , serverhost:serverport)
```

See also, Section G.3.1, "No Oracle WebCenter Portal WLST Commands Work" and Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

G.3.5 More Than One Application with the Same Name Exists in the Domain

Problem

You attempt to perform an operation on WebCenter Portal or a Portal Framework application, such as create a connection for a service or register a portlet producer, and the following message displays:

```
Another application named "YourApplicationName" exists. Specify the Server on
which your application is deployed. Use: server="YourServerName".
```

This message displays if there are multiple applications with the same name in the domain. This usually happens in a cluster environment, where the same application is deployed to multiple managed servers.

For example, you tried to register a portlet producer for an application named "MyApp" using the following WLST command:

```
registerWSRPProducer(appName='myApp', name='MyWSRPSamples',
url='http://myhost.com:9999/ portletapp/portlets/wsrp2?WSDL')
```

Solution

Specify on which managed server you want to run the WLST command, that is, include the `server` argument. For example:

```
registerWSRPProducer(appName='myApp', name='MyWSRPSamples',
url='http://myhost.com:9999/portletapp/portlets/wsrp2?WSDL',
server=WC_CustomPortal2)
```

See also, Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

G.3.6 More Than One Application with the Same Name Exists on a Managed Server

Problem

You attempt to perform an operation on WebCenter Portal or a Portal Framework application such as create a connection for a service or register a portlet producer, and the following message displays:

```
Another application named "application_name" exists on the server
managedServerName.
```

This message indicates that there are multiple applications with the same name on specified managed server. This usually happens when applications are assigned different versions.

For example, you tried to register a portlet producer for an application named "MyApp" using the following WLST command:

```
registerWSRPProducer(appName='myApp', name='MyWSRPSamples',
url='http://myhost.com:9999/portletapp/portlets/wsrp2?WSDL')
```

Solution

Specify on which application version you want to run the WLST command, that is, include the `server` and `applicationVersion` arguments. For example:

```
registerWSRPProducer(appName='myApp', name='MyWSRPSamples',
url='http://myhost.com:9999/portletapp/portlets/wsrp2?WSDL',
server=WC_CustomPortal1, applicationVersion=2)
```

See also, Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

G.3.7 Already in Domain Runtime Tree Message Displays

Problem

While running a WLST command, the following message displays:

```
Already in Domain Runtime Tree
```

Solution

None required. This is for information only.

G.4 Troubleshooting Oracle WebCenter Portal Performance Issues

Use the information in this section to help diagnose performance-related issues for Oracle WebCenter Portal.

This section contains the following sub sections:

- About Performance Monitoring and Troubleshooting Tools
- How to Troubleshoot Overall System Slowness
- How to Identify Slow Pages
- How to Identify Slow Page Components
- How to Troubleshoot Slow Page Requests
- How to Troubleshooting Requests using JRockit Flight Recordings

G.4.1 About Performance Monitoring and Troubleshooting Tools

Various tools are available for monitoring and troubleshooting performance issues with your Oracle WebCenter Portal environment.

Table G–3 Performance Monitoring and Troubleshooting Tools

Tool	Use to...	See
Enterprise Manager		
Fusion Middleware Control	Monitor WebCenter Portal metrics and log files in real-time mode for a single Oracle Fusion Middleware Farm. Check service configuration, including MDS and partitions for WebCenter Portal deployments.	Starting Enterprise Manager Fusion Middleware Control
Grid Control	Monitor WebCenter Portal metrics in real time and from a historical perspective for trend analysis, as well as monitor the underlying host and operating system, databases, and more. Oracle Enterprise Manager 11g Grid Control must be installed separately as it is not a part of the Oracle Fusion Middleware 11g installation. With Grid Control, you can centrally manage multiple Oracle Fusion Middleware Farms and WebLogic Domains.	Oracle Enterprise Manager Grid Control
WebCenter Portal Page Performance Analyzer	Analyze the performance of portal pages in WebCenter Portal. This tool dynamically measures and presents the performance of individual page components when you display pages in WebCenter Portal.	How to Identify Slow Page Components
JConsole	Graphically monitor Java applications and Java virtual machines (JVM).	How to Use JConsole to Monitor JVM
JRockit Mission Control	Capture and present live data about memory, CPU usage, and other runtime metrics.	Troubleshooting Slow Requests Using JFR Recordings

Table G–3 (Cont.) Performance Monitoring and Troubleshooting Tools

Tool	Use to...	See
Eclipse Memory Analyzer	Find memory leaks and reduce memory consumption.	Troubleshooting Memory Leaks and Heap Usage Problems
Threadlogic	Analyze thread dumps.	Generating Thread Dumps to Diagnose Extremely Slow Page Performance, High Thread Counts, and System Hangs

G.4.2 How to Troubleshoot Overall System Slowness

Use the actions listed in Table G–4 to troubleshoot overall system slowness:

Table G–4 Troubleshooting Overall System Slowness

Action	Description	More Information
1	Verify key system resources.	Verifying System Resources (CPU and Memory)
2	Use `top` or `vmstat` to see if system slowest is caused by CPU, memory, or IO contention issues.	Monitoring System Resource Usage
3	Monitor the performance of your Java virtual machine (JVM).	Monitoring Java Virtual Machine (JVM) Usage
4	Verify OID and database connection pool settings.	Verifying Connection Pool Settings
5	Generate automatic workload repository (AWR) reports for Oracle databases to diagnose database-related issues.	Generating Automatic Workload Repository (AWR) Reports for the Database
6	Use `tcpdump` to investigate and diagnose network related problems.	Diagnosing Network Related Problems Using tcpdump
7	Use `ping` to measure network latency.	Measuring Network Latency Using ping
8	Collect thread dumps.	Generating Thread Dumps to Diagnose Extremely Slow Page Performance, High Thread Counts, and System Hangs
9	Look for errors in `WC_Spaces-diagnostic.log` and check the correct logging level is enabled.	Analyzing the Diagnostics Log
10	Use DMS Spy to monitor internal DMS metric data, such as activity in the Java Object Cache.	Using DMS Spy to Monitor Internal Performance Metric Tables
11	Look at the `access.log` for the WebLogic Server to check HTTP request/response cache settings. Verify HTTP compression settings.	Verifying HTTP Request Caching Verifying HTTP Compression
12	Use HTTP monitoring tools to analyze requests and response times.	Checking Browser Response Times
13	Warm up your system before taking performance measurements.	Warm up the System Before Re-Testing Performance

G.4.2.1 Verifying System Resources (CPU and Memory)

If you are experiencing performance issues its important to verify that you have sufficient system hardware resources, that is, adequate CPU and physical memory capacity for your Oracle WebCenter Portal installation.

Low system resources can cause many different problems. System resources are used by individual services. Regularly monitoring and recording system usage can help you determine whether you need to upgrade your system hardware, or if some services need to be moved to another machine.

To verify system CPU and memory:

- **On Linux**, review CPU and memory information in the following files:
  - `/proc/cpuinfo`

    The `cpuinfo` file provides important CPU information including the model, CPU cores, CPU MHz, cache size, and flags which show what instruction sets are available on the processor. Systems with multiple processors or multiple cores have separate entries for each.
  - `/proc/meminfo`

    The important fields to look for in the meminfo file include `MemTotal`, `MemFree`, `Cached`, and `SwapTotal`.
- **On Windows**, access CPU and memory information through the Task Manager (**Performance** > **Resource Monitor**).

G.4.2.2 Monitoring System Resource Usage

On Linux, you can use `top` and `vmstat` utilities to see if system slowness is caused by CPU, memory, or I/O contention issues. If you discover that your system resources are low, you can:

- Move processes to other machines or shut down unused processes/programs to free up more physical memory and/or CPU cycles.
- Upgrade your system hardware resources

For more information, see:

- How to use top to monitor system resource usage on Linux
- How to use vmstat to monitor system resource usage on Linux

> **Note:** On Windows, use Task Manager to monitor system resources and shut down unused processes and programs. Refer to your Windows documentation for more information.

G.4.2.2.1 How to use top to monitor system resource usage on Linux The `top` utility displays a continually updating report of system resource usage so you can identify the top memory and CPU consumers on your system.

The top portion of the report lists information such as the system time, uptime, CPU usage, physical and swap memory usage, and number of processes. Below that is a list of the processes sorted by CPU utilization.

> **Note:** Use `Shift+M` to sort by memory usage. Use `Shift+P` to sort by CPU usage.

```
# top
2:10:49  up 8 day,  3:47,  20 users,  load average: 0.34, 0.19, 0.10
75 processes: 20 sleeping, 2 running, 8 zombie, 0 stopped
CPU states:   5.1% user   1.1% system   0.0% nice   0.0% iowait  93.6% idle
Mem:   512216k av,  506176k used,    6540k free,    0k shrd,   21888k buff
Swap: 1044216k av,  161672k used,  882544k free     199388k cached

PID   USER  PRI NI  SIZE  RSS   SHARE STAT %CPU %MEM   TIME    CPU COMMAND
2330  admin  15  0  161M  70M   2132  S    4.9  14.0   1000m   0    oracle
2605  lin    15  0  8240  6340  3804  S    0.3  1.2    1:12    0    oracle
3413 harvey  15  0  6668  5324  3216  R    0.3  1.0    0:20    0    oracle
```

Troubleshooting Tips - top:

- If `free` memory is < 100Mb and cached memory is < 1GB, system memory is running low.
- If `%wa` (I/O WAIT) is *always* more than 10%, the system may be slow because it is blocked by physical I/O.
- Ensure that the `idle` value is close to 100% and `system/user` CPU usage is close to 0% when there is no load on the system.
- In the memory view of `top`, identify the % memory usage (`%MEM`) for each process (`PID`). If the memory is tight and swap space usage is high, consider:

  - moving process with high memory usage to another machine

  - increasing memory allocation to the virtual machine

  - adding physical memory.

G.4.2.2.2 How to use vmstat to monitor system resource usage on Linux The `vmstat` utility shows running statistics on various parts of the system including system processes, memory, swap, I/O, CPU, and the Virtual Memory Manager. These statistics are generated using data from the last time the command was run to the present. The first time you run the command, data displays from the last reboot until the present.

When you run `vmstat` you can specify how often you want the statistics to refresh (in seconds) using the format: `vmstat <refresh rate in seconds>`

For example: `vmstat 3`

Frequent system resource usage updates, enable you to see trends in CPU/memory usage and how usage trend impact WebCenter Portal or a Portal Framework application performance. If the system is stable, the `swap` metrics (`si` and `so`) should register near zero.

```
# vmstat 3
   procs           memory              swap        io       system      cpu
r  b    swpd  free   buff    cache   si   so    bi    bo   in    cs us sy id wa
3  0     144 1058184 868324 12343056    0    0   220    83   33    14 10  1 88  1
0  0     144 1058184 868324 12343056    0    0     0   117  478   656  5  1 94  0
0  0     144 1058192 868324 12343056    0    0     0    69  696   860  9  1 90  0
2  0     144 1058264 868324 12343056    0    0     0     0  914  1127  5  2 93  0
0  0     144 1057864 868324 12343316    0    0     0   184  857  1021  7  1 92  0
2  0     144 1057592 868324 12343316    0    0     0   155 1596  1646  5  4 91  0
0  0     144 1057592 868324 12343316    0    0     0     0  737   839  5  2 94  0
0  0     144 1057592 868324 12343316    0    0     0    57  694   743  8  2 91  0
2  0     144 1057592 868324 12343316    0    0     0     0  358   437  2  0 98  0
```

Troubleshooting Tips - vmstat:

- When the `swap` metrics (`si` and `so`) are frequently non-zero, system memory becomes tight, and the system may go into thrashing mode. Low system memory significantly affects performance, that is, any program running on the system becomes unusually slow. If you frequently experience low system memory, increase the memory on the machine.
- If CPU idle time (`id`) is constantly less than 10%, the CPU is running at full or over capacity. Under such conditions, any more load on the system results in significant performance degradation.

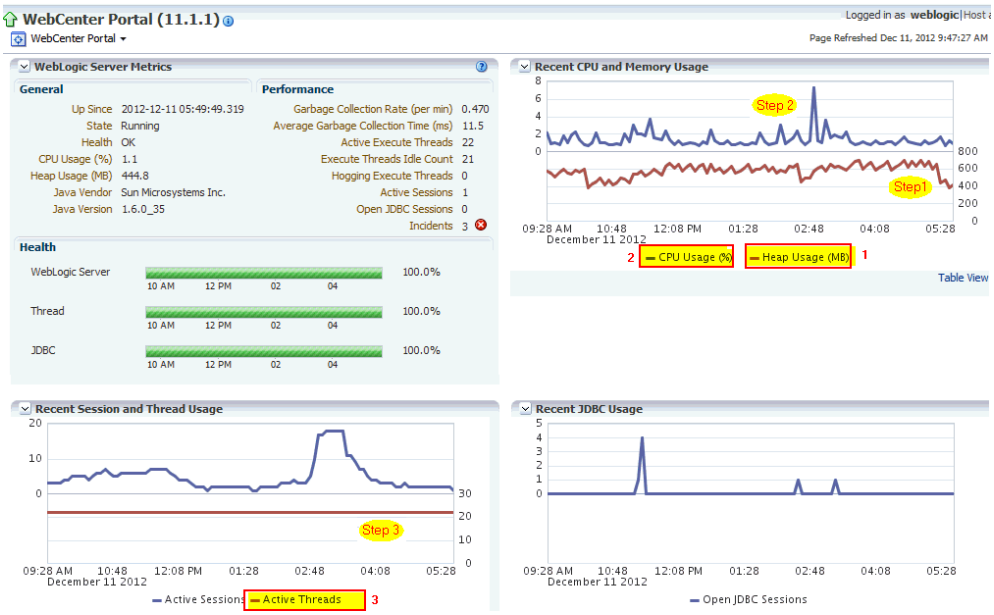
G.4.2.3 Monitoring Java Virtual Machine (JVM) Usage

Your Java virtual machine (JVM) can significantly affect Oracle WebCenter Portal performance. Oracle recommends that you continually monitor your JVM to track:

1. Memory usage
2. CPU usage
3. Thread activity

You can monitor all three metrics from your applications's Recent WebLogic Server Metrics page in Fusion Middleware Control (Figure G–5). For details, see Section 27.1.8, "Understanding WebLogic Server Metrics" and Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

Figure G–5 Recent WebLogic Server Metrics Page - JVM Metrics



- **Monitor memory trends** - JVM frequently allocates and releases memory. To detect memory leaks and high memory usage, you must analyze the memory trend over a long period of time (an hour or several hours).

  When you see the bottom trend line on a memory graph increasing, it indicates a memory leak. The bottom points on a memory graph show how low memory usage can go after full garbage collection. If you take at least two memory dumps, one when the memory is healthy and another when the memory full garbage collection cannot recycle too much, you can compare the memory dumps and identify which components are consuming memory.

See also, Section G.4.5.4, "Troubleshooting Memory Leaks and Heap Usage Problems."

- **Monitor CPU usage** - Occasional spikes in CPU usage is normal but if CPU usage remains high (85-90%) over a long period of time, it normally indicates there is an issue with CPU which can impact performance.

  If CPU usage appears overloaded, comparing several thread dumps at fixed intervals (for example, every 5 seconds) can help reveal causes of high CPU usage.

  Alternatively, profiling tools, such as JRockit's Flight Recorder, can provide further insight into CPU usage.

- **Monitor thread activity and analyze thread dumps** (stuck threads, blocked threads, deadlocks) - The number of threads should stabilize under stable load conditions. Sudden spikes or an increasing number of threads, generally indicates an issue in your system.

  When that is happening, you can take multiple thread dumps to understand the calling stack of stuck/blocked threads or deadlocks. You can use thread analysis tools to identify what is causing the extra threads and what are they doing, as well as detect contentious areas and slow performing areas. See also, Section G.4.5.2, "Troubleshooting Stuck Threads."

  See also, Section G.4.2.8, "Generating Thread Dumps to Diagnose Extremely Slow Page Performance, High Thread Counts, and System Hangs."

  > **Tip:** Oracle Fusion Middleware's Diagnostic Framework collects and manages information about common problems, such as stuck threads and deadlocked threads, to help you diagnose and resolve such issues. Alternatively, you can send diagnostic dumps to Oracle Support Services. For more details, see "Diagnosing Problems" in *Oracle Fusion Middleware Administrator's Guide*.

There are various tools available to monitor JVM performance, including Fusion Middleware Control, JConsole, JVisualVM, and JRockit Mission Control. Use any of these tools to show the current state of the JVM, as well as all the active threads and their states. See also, the "Tuning Java Virtual Machines (JVMs)" section in *Oracle Fusion Middleware Performance and Tuning Guide*

G.4.2.3.1 How to Use JConsole to Monitor JVM

In the first instance, administrators can use the application's Recent WebLogic Server Metrics page in Fusion Middleware Control to monitor JVM (as shown in Figure G–5). If more aggressive real time metrics are required, another option is to use JConsole (shown in Figure G–6).

JConsole is available with all types of JVM, including HotSpot, JRockit, IBM JVM, and so on.

JConsole's executable is located at: `JAVA_HOME/bin/jconsole.exe`.

Figure G–6 JConsole

G.4.2.4 Verifying Connection Pool Settings

This section describes how to verify connection pool settings for:

- WebCenter Portal Data Sources (JDBC Connection Pool Settings)
- Identity Store (JNDI Connection Pool Settings)

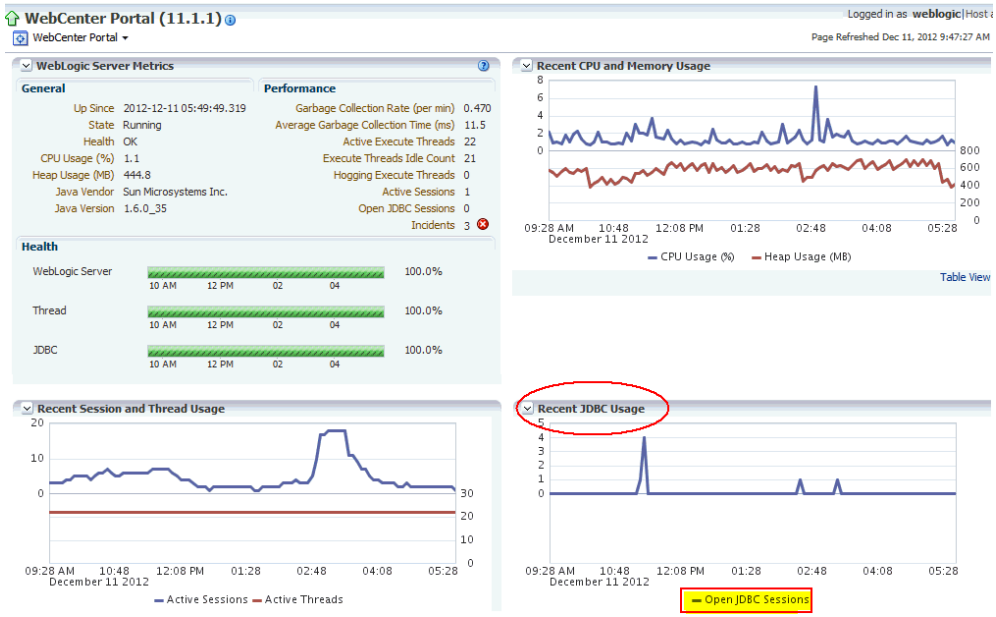
The information in this section might be useful if you are experiencing slow response times and your diagnostics log files contain connection/connection pool messages or a recent thread dump contains calling stacks that are waiting to get connections from connection pool.

G.4.2.4.1 WebCenter Portal Data Sources (JDBC Connection Pool Settings)

Administrators can use Fusion Middleware Control to monitor JDBC connection metrics for WebCenter Portal or a Portal Framework applications. Use the JDBC usage information on the WebLogic Server Metrics Page (Figure G–7) to assess whether JDBC configuration or the connection pool size needs to be adjusted.

See also, Section 27.2, "Viewing Performance Metrics Using Fusion Middleware Control."

Figure G–7 Recent WebLogic Server Metrics Page - JDBC Usage



The Recent JDBC Usage chart shows the number of JDBC connections currently open on the managed server. The JDBC session count that is displayed here, is the sum of the *Current Active Connection Count* metric for each JDBC data source.

If usage is high and the trend is rising, administrators can use WebLogic Server Administration Console to view and configure data source connection pool settings and monitor usage patterns for individual data sources in more detail. If you monitor data source usage over a period of time you can:

- Determine the best connection pool size for a particular database connection.
- Detect and resolve connection pool leakage, that is, if you notice that the number of data source connections do not decrease without load.

Tip: Select "Customize the Table" when monitoring connection pools, to display additional metrics such as:

- Connection Delay Time
- Current Capacity High Count
- Failures To Reconnect Count
- Wait Seconds High Count
- Waiting For Connection Current Count
- Waiting For Connection Failure Total

For high concurrency systems, you may want to adjust the maximum number of connections in the pool (**Maximum Capacity** setting). Out-of-the-box maximum values for the various WebCenter Portal data sources are shown here:

WebCenter Portal Data Sources	Connection Pool Default Maximum Capacity
WebCenterDS	50

WebCenter Portal Data Sources	Connection Pool Default Maximum Capacity
ActivitiesDS	25
mds-SpacesDS	50
mds-owsmDS	15
mds-PageletProducerDS	50
mds-ServicesProducerDS	50
mds-wcpsDS	50
PersonalizationDS	25
Portlet-ServicesProducerDS	50
PortletDS	50
WC-ServicesProducerDS	25

Note: Each connection uses memory in the WebLogic Server and consumes processes in the database so do not specify an unnecessarily large connection pool value.

- To monitor a particular data source, log in to the WebLogic Server Administration Console, select **Services>Data Sources**, click the data source name, and then click the **Monitoring** tab.
- To modify the connection pool for a particular data source, log in to the WebLogic Server Administration Console, select **Services>Data Sources**, click the name of the data source, and then click the **Connection Pool** tab.

See the "Tuning Data Source Connection Pools" section in *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server*.

G.4.2.4.2 Identity Store (JNDI Connection Pool Settings)

Typically, JNDI connection pooling is always turned on. However, some additional configuration is required if the connection between WebCenter Portal/Portal Framework application and the identity store (this may be Oracle Internet Directory, Active Directory, and so on) uses SSL. By default, when you choose an SSL port, the JNDI connections are not pooled causing increased response time and decreased performance when logging in, looking up users, groups, or other identity store entities. For more information and instructions, see the "Tuning Identity Store Configuration" section in *Oracle Fusion Middleware Performance and Tuning Guide*.

G.4.2.5 Generating Automatic Workload Repository (AWR) Reports for the Database

If database access is slow, for example, activity stream queries, MDS queries or JPA (Java Persistence API) queries are slow for various operations in your WebCenter Portal application, you can analyze Automatic Workload Repository (AWR) reports to diagnose the root cause of performance-related problems in your Oracle database.

Before generating the AWR report, first check the general health (CPU/Memory) of that machine hosting your database (see Section G.4.2.2, "Monitoring System Resource Usage"). If system resource limitations are not causing poor performance, examine the database performance information and statistics in the AWR report.

For more detail, see the "Tuning Database Parameters" section in *Oracle Fusion Middleware Performance and Tuning Guide* and the "Generating Automatic Workload Repository Reports" section in *Oracle Fusion Middleware Performance and Tuning Guide*.

G.4.2.6 Diagnosing Network Related Problems Using tcpdump

Use `tcpdump`, a network utility that listeners and captures network traffic to investigate and diagnose network related problems.

For example, if WebCenter Portal page performance metrics in Enterprise Manager indicate that server performance is operating normally while end users are reporting unstable/slow performance, you could have a problem with your network. You can use `tcpdump` or a similar network monitoring tool to trace network your traffic to see if any environmental issue (on the network) is causing an unexpected large latency.

Refer to `tcpdump` documentation for information on how to run this utility and analyze the network data.

G.4.2.7 Measuring Network Latency Using ping

Use `ping` to measure network latency.

Oracle WebCenter Portal installations depend on various backend components and services, such as Oracle HTTP Server, Oracle WebCenter Content Server, LDAP server, instant messaging and presence servers, mail servers, portlets servers, databases, and more. If all your required components are not on the same machine, Oracle recommends they are closely located to minimize network latency and, if possible, are on the same subnet.

Use `ping` from one machine to another to verify that network latency is less than 1ms.

G.4.2.8 Generating Thread Dumps to Diagnose Extremely Slow Page Performance, High Thread Counts, and System Hangs

Oracle recommends that you generate thread dumps to a dedicated thread dump file and then use a thread analysis tool such as ThreadLogic or a simple text editing tool, to understand and correlate the thread execution logic and progress.

To generate thread dumps for Sun JVM (HotSpot) to a file:

```
>jstack <pid> > threaddump.txt
```

To generate thread dumps for Oracle JRockit JVM to a file:

```
>jrcmd print_threads <pid> > threaddump.txt
```

Sometimes its useful to generate a series of thread dumps at fixed sampling intervals to confirm problems such as extremely slow method calls, stuck threads, deadlocks, and so on. For example, you could create a simple script to generate dumps every second or if you are diagnosing a slow request you can choose a suitable duration to cover the length of the slow request.

This example generates thread dumps every second:

```
#!/bin/bash
for i in {1..20}
do
 JAVA_HOME/bin/jstack <pid> > thread_dump_$i.txt
 sleep 1
done
```

For more detail about the capabilities of ThreadLogic or any other thread analysis tool, refer the appropriate manufacturer's documentation.

Note: When WebLogic servers detect a deadlock or stuck thread, a related thread dump is automatically generated in the server's output log file, located at:

DOMAIN_NAME\servers*SERVER_NAME*\logs*SERVER_NAME*.out

For example, the output log for the out-of-the-box application WebCenter Portal is available at DOMAIM_NAME\servers\WC_Spaces\logs\WC_Spaces.out.

The output file is a simple text file that contains various server messages, including thread information.

G.4.2.9 Analyzing the Diagnostics Log

Look for errors, incidents, and warnings in the diagnostics log for the managed server that is hosting WebCenter Portal or a Portal Framework application, that is, <managed_server_name>-diagnostic.log.

When your application is running with some error condition, it can have a big impact on performance. For example, if a connection to WebCenter Content Server becomes intermittent or not accessible, pages with content related components respond very slowing while attempting to connect and eventually may time out.

ERROR, INCIDENT, and WARNING messages due to timeouts, unavailable services, cache errors, and so on, are logged to a diagnostic log file which you can view from Fusion Middleware Control. See also, Section 28.2.1, "Viewing and Configuring WebCenter Portal Logs."

Note: When measuring performance, ensure that DEBUG/TRACE messages, that is, levels lower than CONFIG (700) are not being logged. When FINE, FINER, or FINEST messages display, the system is running in debugging mode and this means that most requests are significantly slower than normal. If you configure lower level logging temporarily to debug a problem, ensure that you change the log level before taking performance measurements.

G.4.2.10 Using DMS Spy to Monitor Internal Performance Metric Tables

The DMS Spy servlet provides access to internal DMS metric data from a web browser. This servlet is useful if you want to monitor the system for long period of time and it can help you understand internal system behavior and performance.

You would not normally use DMS Spy to monitor Oracle WebCenter Portal-specific metrics since this information is more easily available through Fusion Middleware Control. However, if you interested in other underlying metrics, such as warning/error messages about the ADF application module (AM) pool display, you can investigate AM pool metrics here. Similarly, if messages relating to the Java Object Cache (JOC) display, you can turn on the JOC's DMS monitoring feature to observe activities in JOC.

To enable JOC DMS monitoring, edit the javacache.xml file (normally it is in <webcenter_portal_domain>/config/fmwconfig/servers/WC_Spaces/javacache.xml)

1. Open the `javacache.xml` file.

   Typically, the file is located at:
   `<webcenter_domain>/config/fmwconfig/servers/<server_name>/javacache.xml`

   For example, for WebCenter Portal, the file is located at:
   `<webcenter_domain>/config/fmwconfig/servers/WC_Spaces/javacache.xml`

2. Change the entry:

   ```
   <dms enabled="false"/>
   ```

   To:

   ```
   <dms enabled="true"/>
   ```

3. Restart the managed server on which WebCenter Portal or your Portal Framework applications deployed.

   For example, for WebCenter Portal, restart WC_Spaces.

Once enabled, you see two new entries in the left panel: **java_cache_region** and **joc**

java_cache_region shows how much cache memory is allocated to each area and the values displayed can help you determine cache size settings. For example, by default the MDS caching size is set to 100MB in `adf-config.xml`:

```
<max-size-kb>100000</max-size-kb>
```

Both **Region Name: ADFApplication<N>** and **ADFApplication<N>/main_region** are used by the MDS cache. If the sum of **ADFApplication<N>** and **ADFApplication<N>/main_region** memory is close to 100MB, and your JVM has enough unused heap memory, you could increase the MDS caching size (**<max-size-kb>)** so the MDS cache is not so full, and this will improve the efficiency of the MDS cache.

For more information, see the "Viewing Performance Metrics Using the Spy Servlet" section in *Oracle Fusion Middleware Performance and Tuning Guide*.

G.4.2.11 Verifying HTTP Request Caching

Most web pages include resources that do not change very often, such as images, CSS files, JavaScript files, and so on. As such resources take time to download over the network, the time taken to load a web page increases. HTTP caching allows such resources to be saved or *cached*, by a browser or proxy. Once a resource is cached, a browser or proxy can refer to the locally cached copy instead rather than downloading it again on subsequent visits to the web page. Caching reduces round-trip time by eliminating HTTP requests for the required resources. This not only helps significant reducing page load time for subsequent user visits, but also reduces the bandwidth and hosting costs for your site.

By default, static resources serviced by WebCenter Portal use the `ADFCachingFilter` to include the required response headers which allow browsers to cache static content. If your site is experiencing performance issues, you must confirm that your browser is caching static resources. If caching is not correctly working, static WebCenter Portal resources are repeatedly fetched from the server rather than being cached in the browser, and this can impact performance. A possible reason for caching issues with static portal resources could be dues to some loss introduced by a proxy or network component sitting between WebCenter Portal and the browser.

> **Note:** For more information about `ADFCachingFilter`, see *Oracle Fusion Middleware Web User Interface Developer's Guide for Oracle Application Development Framework*.

Use HTTP request monitoring tools, such as Firebug (Firefox) or httpWatch (Internet Explorer and Firefox), to monitor HTTP traffic between the browser and the server. With these tools, you can see if there is a problem with the cache, that is, whether static resources are fetched for each request, not cached for long enough, or not cached at all for static resources (such as JavaScript and CSS files). When using these monitoring tools, confirm that there are browser-cache related tags in the response, for example, `Cache-Control` header with a suitable `max-age` value or an `Expires` tag with a suitable cache expiry time.

You can also analyze HTTP caching by examining the `access.log` for the managed server on which WebCenter Portal or your Portal Framework application is deployed. For example, look at the log to see whether a particular user/IP is repeatedly making requests for the same resource. If the log contains repeated requests, the cache header on requests might be wrong and you must investigate caching issue further. Start by accessing the static resource to eliminate various tiers. For example, use `wget` or `curl` to fetch the static resource directly using the WebLogic server port by issuing the same on the local machine. Next, access the same resource through any other entity front-ending the WebLogic server that is hosting the application, for example Oracle HTTP Serve or Apache. If needed, enable packet tracing to find the response from the WebCenter Portal tier to see whether issues are occurring on the Oracle WebCenter Portal side, or the problem is on the network.

> **Note:** The access log for WebCenter Portal is located at: `ORACLE_HOME/user_projects/domains/wc_domain/servers/WC_Spaces/logs/access.log`

While this section describes static resources serviced by WebCenter Portal, you must also consider similar issues for static resources in your environment. For example, if you have a rich UI where common pages reference static resources that you own, you must review how such content is cached. Consider using Apache `Header` directives to drive the caching of static resources, either based on file type (such as images, CSS), or based on URL path patterns. For example, the following configuration in Apache sends back a response header to the browser to cache all content under the URL path "`/my_images/`" for 30 days (2592000 is the number of seconds in 30 days):

```
<Location /my_images/>
  Header set Cache-Control "max-age=2592000, public"
</Location>
```

G.4.2.12 Verifying HTTP Compression

In addition to HTTP caching, it is important that you review HTTP compression characteristics for responses. HTTP Compression is a publicly defined way to compress content (mostly textual) that is transferred from web servers to browsers. Compression reduces the number of bytes that are transmitted which improves performance. HTTP Compression uses public domain compression algorithms to encode HTML, XML, JavaScript, CSS and other file formats at the server-side. This standards-based method for delivering compressed content is built into HTTP/1.1, and all modern web browsers support the HTTP/1.1 protocol, that is. they can decode

compressed files automatically at the client-side. As a result, no additional software or user interaction is required on the client-side.

By default, static resources serviced by WebCenter Portal use the `ADFCachingFilter()` to compress static content like CSS and JavaScript. If your site is experiencing performance issues, you must confirm that your browser is seeing compressed responses. You can use one of the HTTP request monitoring tool described in Section G.4.2.11, "Verifying HTTP Request Caching" to scan responses for a `Content-Encoding` response header with the value `gzip`. A `Content-Encoding` response header is *only* sent if the client (that is, the browser) supports compressed content.

> **Note:** For more information about `ADFCachingFilter`, see in *Oracle Fusion Middleware Web User Interface Developer's Guide for Oracle Application Development Framework*.

If the browser sends the request header `Accept-Encoding: gzip`, then you know the browser can process compressed content. If you do not see a compressed response, make sure that the client sends an `Accept-Encoding` header stating it can handle compressed content.

File format that typically require compression include: HTML, CSS, XML, and JavaScript. As most images, music, and videos are already compressed, you do not need to configure these file formats for compression.

Consider using Apache's `mod_deflate` or `mod_gzip` to drive the compression of resources, either based on MIME type (such as, text/css) or based on file extension (such as *.html). For example, the following configuration in Apache compresses resources listed in each `AddOutputFilterByType`:

```
<Location />
    SetOutputFilter DEFLATE
    AddOutputFilterByType DEFLATE text/plain
    AddOutputFilterByType DEFLATE text/xml
    AddOutputFilterByType DEFLATE text/html
    AddOutputFilterByType DEFLATE application/xhtml+xml
    AddOutputFilterByType DEFLATE text/css
    AddOutputFilterByType DEFLATE application/xml
    AddOutputFilterByType DEFLATE image/svg+xml
    AddOutputFilterByType DEFLATE application/rss+xml
    AddOutputFilterByType DEFLATE application/atom_xml
    AddOutputFilterByType DEFLATE application/x-javascript
    SetEnvIfNoCase Request_URI \.(?:gif|jpe?g|png)$ no-gzip dont-vary
    SetEnvIfNoCase Request_URI \.(?:exe|t?gz|zip|bz2|sit|rar)$ no-gzip dont-vary
    SetEnvIfNoCase Request_URI \.(?:pdf|doc?x|ppt?x|xls?x)$ no-gzip dont-vary
    SetEnvIfNoCase Request_URI \.avi$ no-gzip dont-vary
    SetEnvIfNoCase Request_URI \.mov$ no-gzip dont-vary
    SetEnvIfNoCase Request_URI \.mp3$ no-gzip dont-vary
    SetEnvIfNoCase Request_URI \.mp4$ no-gzip dont-vary
</Location>
```

G.4.2.13 Checking Browser Response Times

Use HTTP request monitoring tools, such as Firebug (Firefox) or httpWatch (Internet Explorer and Firefox), to monitor HTTP traffic between the browser and the server. If you understand HTTP request flows and timings for common user actions it can help you diagnose and resolve application performance issues. For example, whenever a user logs in, several redirects can happen between the user's browser and servers. If

requests to the login server are taking too long you will know to investigate the login server, rather than WebLogic Server. Similarly, if you see that requests to load the application's landing page is slow, you can focus on making the landing page faster.

G.4.2.14 Warm up the System Before Re-Testing Performance

If you restart an Oracle WebCenter Portal managed server for any reason, make sure that you warm up the system before re-testing performance to avoid Just-In-Time (JIT) compilation overhead.

To warm up your system, you can either manually perform the steps that you later want to measure for performance, or you can use a load test tool to run a few iterations of the steps. After restarting WebLogic managed servers the initial requests are often slower than usual, that is, not typical of the performance that end users would experience.

G.4.3 How to Identify Slow Pages

Use Fusion Middleware Control to determine the slowest pages in a WebCenter Portal or a Portal Framework application. If a poorly performing page is also very popular (the **Invocation** metric is high) then it makes sense for you to focus efforts to improve performance on those pages.

To find the slowest pages:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal or Portal Framework application:
   - Section 6.2, "Navigating to the Home Page for WebCenter Portal"
   - Section 6.3, "Navigating to the Home Page for Portal Framework Applications"
2. Do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Monitoring** > **Recent Page Metrics**.
   - For WebCenter Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Recent Page Metrics**.

   Page requests that respond slower than the `pageResponseTime` threshold display "red" in the chart at the top of the page.
3. Click the **Sort Descending** arrow in the **Time (ms)** column to sort the page requests by response times.

   Page response times that exceed the threshold display "orange" in the table.
4. Identify the slowest pages and make a note of the portal in which the page displays.
5. For more detailed metrics, including how frequently the slowest pages are requested, do one of the following:
   - For WebCenter Portal - From the **WebCenter Portal** menu, select **Monitoring** > **Overall Page Metrics**.

     Note: Requests for pages in the Home portal are excluded from the "Overall Page Metrics" page.
   - For Portal Framework applications - From the **Application Deployment** menu, select **WebCenter Portal** > **Overall Page Metrics**.

See also, Section 27.1.5, "Understanding Page Request Metrics" and Section 27.3, "Customizing Key Performance Metric Thresholds and Collection."

G.4.4 How to Identify Slow Page Components

Use WebCenter Portal's page performance analyzer to quickly see how long individual components take to display on a portal page, as well as the overall time taken to display a page. When enabled, this tool dynamically measures and presents the performance of individual page components whenever you display a portal page.

The portal page performance analyzer is useful to developers who are performing first level performance analysis, customers who build their own pages, and any user who customizes pages in WebCenter Portal.

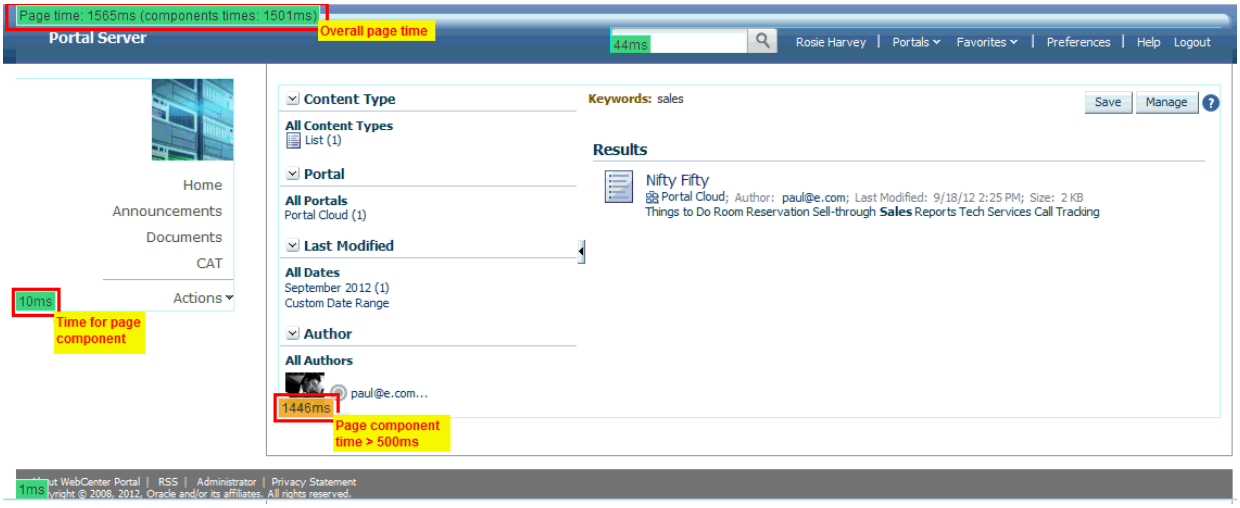
This section includes the following subsections:

- About the Portal Page Performance Analyzer
- Enabling and Disabling Portal Page Performance Analysis
- Displaying and Hiding Page Timing Information for Your Current Session
- Using the Page Performance Analyzer to Troubleshoot Performance Issues
- Limitations

G.4.4.1 About the Portal Page Performance Analyzer

The portal page analyzer offers a simple way to diagnose slow pages and requires minimal set up or configuration. When this feature is on, the time spent on "high level" page components is calculated and displayed so you can see at a glance which components are slowing down your page. The overall time spent on the page also displays at the top left of the page (see Figure G–8).

Figure G–8 Portal Page Displays Timing Information



About Page Component Timings

In WebCenter Portal, "high level" page components are wrapped in a *ShowDetailFrame* so they can be moved, hidden or shown on the page, and edited by Oracle Composer and it is the overall timing for each ShowDetailFrame that displays.

About Overall Page Time

The overall page time is the sum of the individual page component timings, plus some additional processing time for page-level operations such as session replication, save and restore page state, page level security checks, and so on.

Color Coding

Performance timings display in various colors to help alert you to problem areas. Refer to the table below:

Color	Time to Display
Green	< 100 ms
Green/Yellow	100 - 500 ms
Yellow	500 ms - 1 second
Orange	1 - 3 seconds
Red	> 3 seconds

G.4.4.2 Enabling and Disabling Portal Page Performance Analysis

The portal page performance analyzer is disabled out-of-the-box. To make use of this feature, an administrator must specifically enable its use; while the impact on page performance to run this tool is minimal some additional page processing is required.

In a production environment, Oracle recommends that the analyzer is generally disabled to avoid the additional performance data collection and processing and then dynamically enabled when someone reports performance issues for a particular page.

If you do not want end users to see performance data in a production environment, this is another reason to disable the analyzer most of the time.

To enable or disable the portal page analyzer for a WebCenter Portal instance:

1. Use the WLST command `exportMetadata` to export the base `webcenter-config.xml` file from MDS.

   For example:

   ```
   exportMetadata(application='webcenter', server='WC_Spaces',
   toLocation='/tmp/mydata',
   docs='/oracle/webcenter/webcenterapp/metadata/webcenter-config.xml')
   ```

2. Open `webcenter-config.xml` exported from MDS in a text editor and set the `perfdebug-enabled` attribute to `true` to enable or `false` to disable this feature.

   For example:

   ```
   <webcenter:perfdebug-enabled>true</webcenter:perfdebug-enabled>
   ```

3. Save and close `webcenter-config.xml`.
4. Import the updated `webcenter-config.xml` file to MDS.

   For example:

   ```
   importMetadata(application='webcenter', server='WC_Spaces',
   fromLocation='/tmp/mydata',
   docs='/oracle/webcenter/webcenterapp/metadata/webcenter-config.xml')
   ```

There is no need to restart WebCenter Portal to effect this change.

Page performance information does not automatically display after you enable this feature. Anyone who wants to see timing information on portal pages must specifically request that the information displays. For details, see Section G.4.4.3, "Displaying and Hiding Page Timing Information for Your Current Session."

G.4.4.3 Displaying and Hiding Page Timing Information for Your Current Session

When an administrator enables the page performance analyzer in an WebCenter Portal instance, anyone with access to that WebCenter Portal instance can elect to display or hide page timing information, for their current user session, by appending the `perfDebug` parameter to the page URL as follows:

To...	Add a perfDebug parameter to the page URL
Display timing information on portal pages	`&perfDebug=on`
Stop displaying page performance information	`&perfDebug=off`

To display timing information on portal pages:

1. Verify that your administrator enabled the page performance analyzer in your WebCenter Portal instance.

   See also Section G.4.4.2, "Enabling and Disabling Portal Page Performance Analysis."

2. Log in to WebCenter Portal and navigate to the portal page that you want to investigate.

   You do not need to log in if the page is a public page.

3. Add `&perfDebug=on` to the end of the page URL (Figure G–9).

   For example:

   ```
   http://mycompany.com/webcenter/portal/MySalesPortal?_adf.ctrl-state=mji8314i6_4
   &_afrLoop=474789135539036&perfDebug=on
   ```

4. Click "Go" or press Enter to redisplay the page with timing information (as shown in Figure G–8).

   All subsequent pages that you display show timing information as well.

Figure G–9 Appending the perfDebug Parameter to Page URLs

To stop displaying page timing information:

1. In your browser, add `&perfDebug=on` to the end of any page URL (Figure G–9).

   For example:

   ```
   http://mycompany.com/webcenter/portal/MySalesPortal?_adf.ctrl-state=mji8314i6_4
   &_afrLoop=474789135539036&perfDebug=off
   ```

2. Click "Go" or press Enter to display the page again without timing information.

G.4.4.4 Using the Page Performance Analyzer to Troubleshoot Performance Issues

The steps in this section describe how to troubleshoot slow pages using WebCenter Portal tools:

1. If a user reports performance issues with a particular page, navigate to the slow pages and confirm that the slow performance consistently reproduces.

   Alternatively, use Fusion Middleware Control to proactively identify the slowest pages in your application. See Section G.4.3, "How to Identify Slow Pages."

2. Append `&perfDebug=on` to the page URL to display timing information for the page.

   See also, Section G.4.4.3, "Displaying and Hiding Page Timing Information for Your Current Session."

   > **Note:** If page timing information does not display, ask your administrator to enable the page performance analyzer. For details, see Section G.4.4.2, "Enabling and Disabling Portal Page Performance Analysis."

3. Identify the slowest page components, and troubleshoot the issue further:

   For example, if the slow component contains:

   - **Document, wiki, or content presenter**, check the performance of the back-end Content Server and the database that Content Server is using.
   - **Activity stream**, use AWR reports to check database performance and to see if you can tune the database table used by activity stream.
   - **Collaboration features**, check the performance of the associated back-end server. For example, for announcements or discussions, monitor the performance of the discussions server.
   - **Portlets**, use Fusion Middleware Control to monitor portlet request timing information, errors, portlet producer performance, and so on

4. If necessary, add the slow page component to a separate "blank" page and then do further profiling.

   For example, use JRockit flight recording to pinpoint the bottleneck.

G.4.4.5 Limitations

The page performance analyzer works best with portal pages that you create with WebCenter Portal 11.1.1.8.0 and later. In previously releases, some page templates did not include the necessary tags to display component and total page timings and some task flows were not configured with `activation=deferred`.

G.4.5 How to Troubleshoot Slow Page Requests

Use the information in this section to diagnose issues relating to poor page performance:

- Troubleshooting Live Requests
- Troubleshooting Stuck Threads

- Troubleshooting Slow Requests Using JFR Recordings
- Troubleshooting Memory Leaks and Heap Usage Problems
- Troubleshooting Slow Requests for Content

G.4.5.1 Troubleshooting Live Requests

To troubleshoot slow page requests that are still running, extract and view a JRockit Flight Recorder (JFR) recording against the server on which the user session is running. See also, Section G.4.6, "How to Troubleshooting Requests using JRockit Flight Recordings."

Alternatively, take several evenly spaced thread dumps, for example, every 1 or 2 seconds as described in Section G.4.2.8, "Generating Thread Dumps to Diagnose Extremely Slow Page Performance, High Thread Counts, and System Hangs."

If you compare the thread dumps, you might see threads that spent a long time on certain method calls as the call stacks are the same in several consecutive thread dumps. For example, you might see a method call to a database, Oracle WebCenter Content Server, collaboration server, portlet producer, LDAP server, and so on, in which case you can investigate the associated backend server to diagnose the issue further.

G.4.5.2 Troubleshooting Stuck Threads

Stuck threads can occur for several reasons:

- **Server is nearly out of memory.** If the server is close to out of memory, all requests slow down. To resolve out-of-memory issues, see Section G.4.5.3, "Troubleshooting Slow Requests Using JFR Recordings."
- **Deadlock threads.** Take thread dumps and search for deadlock threads. This normally exposes an issue with the product code.
- **Extremely slow page requests.** Take several evenly spaced thread dumps and find out which method is taking a long time to execute.

  If a request is taking longer than 10 minutes, the stuck thread is reported to Oracle WebLogic Server *server_name*`.out` in the following directories:

  ```
  (UNIX) DOMAIN_HOME/servers/server_name/logs
  (Windows) DOMAIN_HOME\servers\server_name\logs
  ```

  For example:

  ```
  <Mar 4, 2012 7:44:08 AM PST> <Error> <WebLogicServer> <BEA-000337>
  <[STUCK] ExecuteThread: '19' for queue: 'weblogic.kernel.Default (self-tuning)'
   has been busy for "600" seconds working on the request
   "weblogic.servlet.internal.ServletRequestImpl@18986012[

  GET

  /server_name/faces/PimDashboardUiShellPage?_afrLoop=1398820150000&_afrWindowMod
  e=0&_adf.ctrl-state=a44e7uxcc_13 HTTP/1.1
  Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg,
   application/x-shockwave-flash, application/x-ms-application,
   application/x-ms-xbap, application/vnd.ms-xpsdocument, application/xaml+xml,
   application/vnd.ms-excel, application/vnd.ms-powerpoint, application/msword,
   */*
  Accept-Language: fr
  UA-CPU: x86
  ...
  ```

```
]", which is more than the configured time (StuckThreadMaxTime) of "600"
 seconds
. Stack trace:
Thread-164 "[STUCK] ExecuteThread: '19' for queue: 'weblogic.kernel.Default
 (self-tuning)'" <alive, in native, suspended, priority=1, DAEMON> {
    jrockit.net.SocketNativeIO.readBytesPinned(SocketNativeIO.java:???)
    jrockit.net.SocketNativeIO.socketRead(SocketNativeIO.java:24)
    java.net.SocketInputStream.socketRead0(SocketInputStream.java:???)
    java.net.SocketInputStream.read(SocketInputStream.java:107)
...
```

Diagnosing a Stuck Thread

If the stack shows the thread is waiting for a response from another server, check the status of the other server and see it has performance problems before proceeding with the steps below.

To determine what the stuck thread was doing prior to becoming stuck, perform the following steps:

1. Look at the next few log messages in *server_name*`.out` for a message indicating an incident has been created. For example:

   ```
   <Mar 4, 2012 7:44:10 AM PST> <Alert> <Diagnostics> <BEA-320016>
   <Creating diagnostic image in DOMAIN_HOME/servers
   /server_name/adr/diag/ofm/MyDomain/
   server_name_1/incident/incdir_394 with a lockout minute
   period of 1.>
   ```

   The above message may not always appear after each stuck thread reported. It is printed at most four times an hour. If the message does not appear, manually look for the `incident` directory by checking the `readme` file in the subdirectories under the following directories:

   ```
   (UNIX)
   DOMAIN_HOME/servers/server_name/adr/diag/ofm/domain_name/server_name/incident
   (Windows)
   DOMAIN_HOME\servers\server_name\adr\diag\ofm\domain_name\server_name\incident
   ```

   The incident directory contains a WLDF diagnostic image which contains the JFR recording, and a file containing the thread dump.

   For more information about diagnosing incidents, see the "Diagnosing Problems" chapter in the *Oracle Fusion Middleware Administrator's Guide*.

2. Review the thread dump to find the call stack of the thread. If the thread is blocked waiting for a lock, check what the thread holding the lock is doing.

3. If the call stack shows that JDBC calls are taking a long time, generate an AWR report on the database to find the query and which table to look and tune.

   See Section G.4.2.5, "Generating Automatic Workload Repository (AWR) Reports for the Database."

4. Review the JRockit flight recording file `JRockitFlightRecorder.jfr` for more details. You will also need the ECID of the request which is recorded in the `readme.txt` file of the incident directory, and also the Oracle WebLogic Server log.

   See also, Section G.4.6, "How to Troubleshooting Requests using JRockit Flight Recordings."

The ECID of the request that caused the stuck thread is recorded in the error message.

G.4.5.3 Troubleshooting Slow Requests Using JFR Recordings

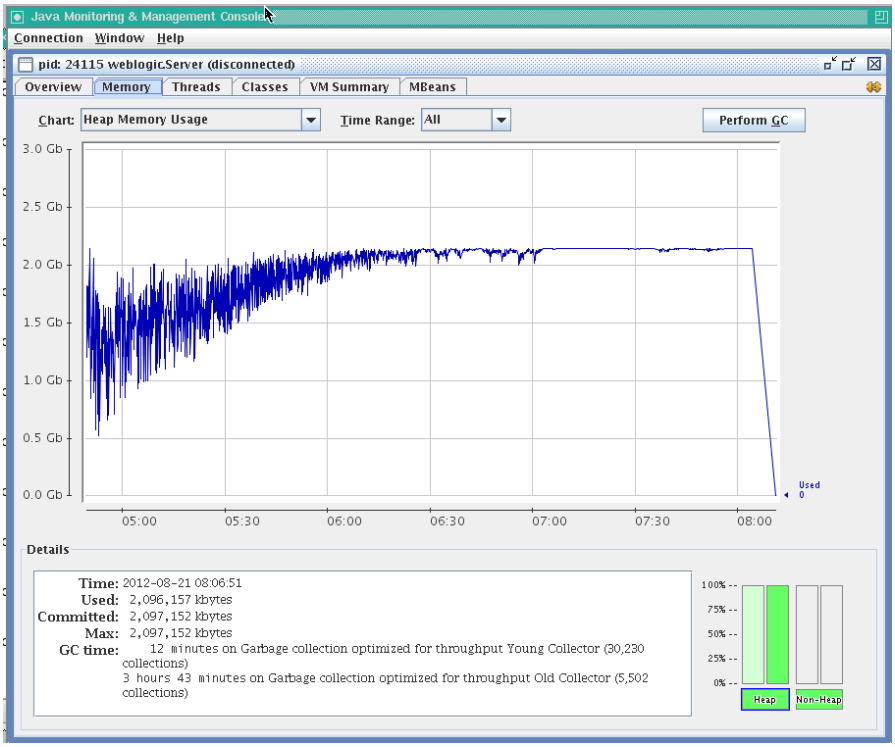
See Section G.4.6, "How to Troubleshooting Requests using JRockit Flight Recordings."

G.4.5.4 Troubleshooting Memory Leaks and Heap Usage Problems

If WebCenter Portal or Portal Framework application performance degrades over time, heap usage and garbage collection activity is increasing, and you see `OutOfMemoryErrors`, there could be memory leaks in the application causing the amount of free memory in the JVM to continuously decrease.

Figure G–10 shows a typical memory leak trend displayed in JConsole.

Figure G–10 Typical Memory Leak Trend in JConsole



To solve this problem:

1. Determine the cause of `OutOfMemoryErrors` errors:
   - Review the `server_name.out` file for `OutOfMemoryErrors` errors.

     The `server_name.out` file is located at:

     `(UNIX)` *`DOMAIN_HOME`*`/servers/`*`server_name`*`/logs`
     `(Windows)` *`DOMAIN_HOME`*`\servers\`*`server_name`*`\logs`

   - Take a memory dump when `OutOfMemoryErrors` errors occur.

     For example:

     On Sun HotSpot: `jmap -dump:live,format=b,file=<path>/heap.hprof <pid>`

     On JRockit: `jrcmd <pid> hprofdump filename=<path>/heap.hprof`

     You can configure JRockit to automatically generate a heap dump in HPROF binary format (`.hprof` file) each time an `OutOfMemoryErrors` occurs, by

setting the JRockit JVM option, `-XX:+HeapDumpOnOutOfMemoryError`. For details, refer to the *Oracle JRockit Command-Line Reference*.

2. Restart the managed server.

   If the problem persists, proceed to Step 3.

3. Open the `heap.hprof` file with a heap-dump analysis tool that can handle binary `HPROF` format, such as Eclipse Memory Analyzer.

4. Determine which objects and classes are retaining the most memory.

5. If necessary, take several heap dumps to determine which objects or classes are consuming and increasing the amount of memory.

   Take at least two memory dumps:

   - Take the first dump when the system is warmed up and stabilized.
   - Take the second dump, when the system is about to run out of memory, that is, full garbage collection gets less than 300MB from the maximum heap size.

   Instructions on how to take a heap dump using Sun HotSpot (`jmap`) or JRockit (`jcrcmd`) is described in step 1.

   See the "Running Diagnostic Commands" chapter in the *Oracle JRockit JDK Tools Guide*. Many heap dump analysis tools, such as Eclipse Memory Analyzer, enable you to compare two heap dumps to identify memory growth areas.

   Heap dumps provide information on why memory is retained (Retained Heap). Sometimes it is necessary to know how memory is allocated to further resolve the issue. For these cases, proceed to Step 6.

6. Use the JRockit Memory Leak Detector tool that is part of JRockit Mission Control Client to understand how memory is allocated.

   For more information, see the JRockit Mission Control online help.

G.4.5.5 Troubleshooting Slow Requests for Content

If slow page performance is due to content/document-related components, for example, Documents service task flows, Content Presenter task flows, wikis or blogs, Oracle recommends that you review performance metrics for the backend Oracle WebCenter Content Server (System Audit Information page). For details, see the "Viewing System Audit Information" section in *Oracle Fusion Middleware Administering Oracle WebCenter Content*.

Ensure that the **systemdatabase** tracing option is selected so you can see performance information for each query that is sent to the database. For details, see the "Server-Wide Tracing" section in *Oracle Fusion Middleware Administering Oracle WebCenter Content*.

G.4.6 How to Troubleshooting Requests using JRockit Flight Recordings

JRockit Flight Recorder (JFR) files contain a record of various events that consume time. If requests are slow, you can analyze the JRockit Flight Recorder (JFR) file to find out why request are taking time.

To create a JFR file:

1. Extract a JFR file from theOracle WebLogic Server server by running the following command:

   ```
   UNIX) JROCKIT_HOME/bin/jrcmd jrockit_pid dump_flightrecording recording=1
   ```

```
copy_to_file=path compress_copy=true

(Windows) JROCKIT_HOME\bin\jrcmd.exe jrockit_pid dump_flightrecording
recording=1 copy_to_file=path compress_copy=true
```

For more information about the `jrcmd` command-line tool, see the "Running Diagnostic Commands" section in *Oracle JRockit JDK Tools Guide*.

2. To view the file, start the JRockit Mission Control Client from the following directories:

```
(UNIX) JAVA_HOME/bin/bin/jrmc

(Windows) JAVA_HOME\bin\jrmc.exe
```

3. Select **File** > **Open File** to select the JFR file.
4. Locate the slowest requests or investigate a specific request:

To locate the slowest requests:	To investigate a specific request:
1. In the JRockitFlightRecorder.jfr page, click the **Events** icon. 2. Click the **Log** tab at the bottom of the page. 3. In the **Event Type** navigation pane on the left, locate **Dynamic Monitoring System** and then **HttpRequest**. 4. Click **HTTP request**; de-select all the other event types. 5. In the **Log** tab, in the **Event Log** section, click the **Duration** column to sort the duration in descending order. Each row corresponds to a HTTP Request and the duration column shows the response time for that request. 6. Click the row in the table to view the attributes of the requests. 7. In the **Event Attributes** sections, note the start time and the thread that serviced the request.	1. Find the Execution Context Identifier (ECID) of that request. If the request is related to an incident triggered by a `STUCK` thread, the incident `readme.txt` file will contain the ECID. Alternatively, you can search the Oracle WebLogic Server HTTP `access.log` for requests from specific users. See the "Viewing and Searching Log Files" section in the *Oracle Fusion Middleware Administrator's Guide.* 2. In the JRockit Mission Control Client, in the JRockitFlightRecorder.jfr page, select the **WebLogic** icon. Note: If the **Weblogic** icon is not available, select **Help** > **Install Plugins** to download the Oracle WebLogic Server plug-in. 3. Click the **ECIDs** tab at the bottom of the page. 4. In the **ECIDs** section, from **Filter Column** list, select **ECID**. 5. Enter the ECID in the search box and select <Enter>. 6. In the results table, highlight the row with the matching ECID and right-click to bring up the menu. 7. Select **Operative Set** > **Clear**, and then **Operative Set** > **Add matching ECID** > *ECID* to add the ECID to the operative set. This enables users to view only events associated with the operative set. 8. Click the **Events** icon. 9. In the Event Type navigation pane on the left, locate **Dynamic Monitoring System** and then **HttpRequest**. 10. Click **HTTP request**; de-select all the other event types. ** 11. In the **Event Log** section, click **Show Only Operative Set**. Each row corresponds to the request with the matching ECID. 12. Click the row in the table to view the attributes of the requests. 13. Note the start time and the thread that serviced the request.

5. Once you have identified the start time and the thread that serviced the request, navigate to the **Logs** tab, and drag the time selector at the top of the screen to include only the time window for the duration of the request.
6. In the **Event Log** section, perform the following search:
   a. Deselect **Show Only Operative Set**.
   b. Enter the thread name in the search box.
   c. From the **Filter Column** list, select **Thread**.

d. Select <Enter>.

7. In the **Event Type** navigation pane on the left, click the events of interest. Typically, these events are located under nodes **Dynamic Monitoring System**, **Java Application**, and **WebLogic** > **JDBC**.

   The selected events appear in the table in the **Event Log** section.

8. Click the **Start Time** column to sort the time when these events occur, or click the **Duration** column to view the events that took longest.

   The **JDBC Statement Execute** events corresponds to SQL execution. If there are slow SQL statements, the event details give the SQL text. These events do not have callstacks.

9. To check the call stacks for slow SQL statements, view the **Socket Read** event that happens immediately after the **JDBC Statement Execute** event.

   This event corresponds to Oracle WebLogic Server waiting for the SQL results to return, and it has callstack in the event details.

10. Review the call stacks for long **Java Blocked** and **Java Wait** events to see if you can identify what is causing slow performance.

    See the "Analyzing Flight Recorder Data in JRockit Mission Control" section in *Oracle Fusion Middleware Configuring and Using the Diagnostics Framework for Oracle WebLogic Server*.

11. If you need more detail than the information captured in the default recording, and you can reproduce the slow requests, you can start an explicit recording.

    See the "Starting an Explicit Recording" section in *Oracle JRockit Flight Recorder Run Time Guide*.

G.5 Using My Oracle Support for Additional Troubleshooting Information

You can use My Oracle Support (formerly MetaLink) to help resolve Oracle WebCenter Portal problems. My Oracle Support contains several useful troubleshooting resources, such as:

- Knowledge base articles
- Community forums and discussions
- Patches and upgrades
- Certification information

> **Note:** You can also use My Oracle Support to log a service request.

You can access My Oracle Support at `https://support.oracle.com`.

G.6 Troubleshooting WebCenter Portal Workflows

If you experience issues with WebCenter Portal workflows, review the following sections:

- Section G.6.1, "Validating the WebCenter Portal Workflow Configuration"
- Section G.6.2, "Troubleshooting Issues with WebCenter Portal Workflows"

G.6.1 Validating the WebCenter Portal Workflow Configuration

The *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal* describes how to install and configure WebCenter Portal workflows. For details, see the "Back-End Requirements for WebCenter Portal Workflows" section. You can validate the workflow configuration as follows:

1. Log in to WebCenter Portal.
2. Create a portal and then navigate to the **Members** tab (click the **Administration** link, then **Security**, then **Members**).
3. Invite a new member with any role (say `User2`).
4. Log out, and then log in as `User2`.
5. Navigate to a **Worklist** task flow.
6. Open the invite notification and click the **Accept** button.
7. Open the portal.

If the WebCenter Portal workflows are working properly, the newly created portal ia available to `User2`. If the portal is not available or listed, there is some issue with the configuration.

G.6.2 Troubleshooting Issues with WebCenter Portal Workflows

If WebCenter Portal workflows are not working properly, follow these steps to help troubleshoot the issue:

1. Check that WebCenter Portal workflows are deployed on the Oracle SOA server:
   a. Log in to Fusion Middleware Control.
   b. Check that `WebCenterWorklistDetailApp.ear` is deployed.
   c. Verify that `sca_CommunityWorkflows.jar` is deployed.

   For details, see the "Oracle SOA Server - Extending the Domain" section in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*.
2. Ensure the Web Service connection between the Oracle SOA server and the WebCenter Portal application is secure:
   a. Check the alias in the keystore file on the Oracle SOA server.

      For example, use the following command to list the content of the keystore file on the Oracle SOA server:

      ```
      keytool -list -v -keystore bpel.jks -storepass <password>
      ```

      There should be an entry with:

      ```
      Alias name: webcenter_portals_ws
      ```

      See the "Setting Up the SOA Domain" section in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*.
   b. Verify that the credential stores for both WebCenter Portal and Oracle SOA server are configured correctly.
   c. Check that keystores exist at both ends of the connection, for example:

      - `webcenter.jks` (copied to WebCenter Portal server end)

      - `bpel.jks` (copied to Oracle SOA server end)

For example, the following commands generate `webcenter.jks` and `bpel.jks`:

```
keytool -genkeypair -keyalg RSA -dname
"cn=webcenter,dc=us,dc=oracle,dc=com" -alias webcenter -keypass mypassword
-keystore webcenter.jks -storepass mypassword -validity 360
keytool -exportcert -v -alias webcenter -keystore webcenter.jks -storepass
mypassword -rfc -file webcenter.cer
keytool -importcert -alias webcenter_spaces_ws  -file webcenter.cer
-keystore bpel.jks -storepass mypassword
keytool -genkeypair -keyalg RSA -dname "cn=bpel,dc=us,dc=oracle,dc=com"
-alias bpel -keypass mypassword -keystore bpel.jks -storepass mypassword
-validity 360
keytool -exportcert -v -alias bpel -keystore bpel.jks -storepass mypassword
-rfc -file bpel.cer
keytool -importcert -alias bpel -file bpel.cer -keystore webcenter.jks
-storepass mypassword
```

See the "Creating the SOA Domain Keystore" section in *Oracle Fusion Middleware Administering Oracle WebCenter Portal*.

d. Configure role members for the `BPMWorkflowAdmin` application role in Oracle SOA server (`soa-infra`).

When associating the domain with an identity store that does not contain the user `weblogic`, you must assign some other valid user to the application role `BPMWorkflowAdmin`. Use WLST commands to do this from the SOA Oracle home, for example, to assign a user named "monty" that exists in LDAP:

```
cd $SOA_ORACLE_HOME/common/bin/
wlst.sh

connect('<admin username>','<admin password>',
'mysoahost.example.com:7001')
revokeAppRole(appStripe="soa-infra", appRoleName="BPMWorkflowAdmin",
principalClass="oracle.security.jps.service.policystore.ApplicationRole",
principalName="SOAAdmin")
grantAppRole(appStripe="soa-infra", appRoleName="BPMWorkflowAdmin",
principalClass="weblogic.security.principal.WLSUserImpl",
principalName="monty")
```

See the "Overview of WLST Security Commands" section in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

G.7 Troubleshooting WebCenter Portal Import and Export

This section contains the following subsections:

G.7.1 ResourceLimitException Issue

Problem

The `ResourceLimitException` error displays when you try to export all portals or the entire WebCenter Portal instance:

```
Weblogic.common.resourcepool.ResourceLimitException
```

Solution

Increase the maximum capacity in the JDBC connection pool. To reconfigure the connection pool, log in to the WLS Administration Console. From **Services**, select **Data Sources**, **WebCenterDS**, and then the **Connection Pool** tab.

G.7.2 LockRefreshTask Issue

Problem

A `LockRefreshTask` warning displays similar to that below when you try to import or export an entire WebCenter Portal instance or a portal:

```
[WARNING] [][oracle.webcenter.lifecycle.operation.LockRefreshTask]
```

If you try the import or export operation again, an error similar to that shown here displays:

```
Starting WebCenter Portal application import...
WebCenter Portal application import started.

Error occurred while performing import
None
Check the WebCenter Portal log files for additional details.
Unable to contact MBeanServer for
oracle.webcenter.lifecycle:ApplicationName=webcenter,Location=WC_Spaces,name=Lifec
ycleManager,type=LifecycleManager,Application=webcenter,ApplicationVersion=11.1.1.
4.0
Error occurred while destroying MBean
The lock hasnt been released from the previous failed import.
```

Solution

Use the `deleteMetadata` WLST command to delete unwanted locks in MDS that may be created and not destroyed due to an unexpected and unusual import or export operation failure. Depending on the operation that failed, run one of the following commands:

For WebCenter Portal application import failure, run:

```
deleteMetadata(application='webcenter', server='WC_Spaces',
docs='/oracle/webcenter/lock/applicationImport/applicationImport.xml')
```

For WebCenter Portal application export failure, run:

```
deleteMetadata(application='webcenter', server='WC_Spaces',
docs='/oracle/webcenter/lock/applicationExport/applicationExport.xml')
```

For portal import failure, run:

```
deleteMetadata(application='webcenter', server='WC_Spaces',
docs='/oracle/webcenter/lock/scopeImport/**')
```

For portal export failure, run:

```
deleteMetadata(application='webcenter', server='WC_Spaces',
docs='/oracle/webcenter/lock/gsExportImport/**')
```

G.7.3 Portals and Portal Templates Not Available After Import

Problem

When you first log in to WebCenter Portal after the import operation, the portals and portal templates that you migrated are not available as expected. This can sometimes occur if the portal or portal template cache fails to refresh properly.

Solution

Refresh the portal or portal template cache manually using the `refreshGroupSpaceCache` and `refreshSpaceTemplateCache` WLST commands.

To completely clear the cache (all portals):

```
refreshGroupSpaceCache(appName='webcenter', spaceNames='', syncMode=1,
updateType='all', cleanCache=1)
```

To completely clear the cache (all portal templates):

```
refreshSpaceTemplateCache(appName='webcenter', spaceTemplateNames='',
syncMode=1, updateType='all', cleanCache=1)
```

For detailed command syntax and examples, see the "refreshGroupSpaceCache" and "refreshSpaceTemplateCache" sections in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

For information on how to run WLST commands, see Section 1.13.3.1, "Running Oracle WebLogic Scripting Tool (WLST) Commands."

G.7.4 Unable to Migrate Portals or Documents If the Source and Target Applications Share the Same Content Server

You cannot migrate portals or portal templates between two different WebCenter Portal instances that share the same Content Server.

G.8 Troubleshooting Individual Portal and Portal Template Import and Export

This section contains the following subsections:

- Section G.8.8, "Importing from the Subportals Page"
- Section G.8.9, "Unable to Import a Portal If the Source and Target Applications Share the Same Content Server"
- Section G.8.10, "Exporting and Importing Portals in Multibyte Languages"

G.8.1 Portal Blocked After Unsuccessful Export or Import

If an error occurs during a portal export/import operation, some portals may appear blocked. To unblock a portal, bring the portal back online temporarily, and then take the portal offline again to complete the export/import operation. Switching between the online and offline modes will unblock the portal. For more information, see Section 46.11, "Taking Any Portal Offline" and Section 46.12, "Bringing Any Portal Back Online." See also, the WLST command "`setSpaceState`" in *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

G.8.2 Page or Portal Not Found Message After Import

When users first log in to WebCenter Portal after an import operation, they may see a "Page not found" or "Portal not found" message if the page or portal they last visited no longer exists. Last accessed page information is retained during import operations which is why these messages display sometimes.

G.8.3 Portal Import Archive Exceeds Maximum Upload File Size

Problem

There is a file size limitation uploading content to WebCenter Portal. If your export archive exceeds the maximum upload size, the import operation through WebCenter Portal Administration will fail.

Solution

Import the portal archive using WLST. For details, see Section 40.1.2.4.3, "Importing a Portal from an Archive Using WLST."

Alternatively, modify the maximum upload size in `webcenter-config.xml`. The default maximum upload size is 2 GB. See Section 9.12, "Changing the Maximum File Upload Size."

G.8.4 Maximum Number of Portals Exceeded on Export

Problem

The maximum number of portals that you can export must be less than or equal to 80% of the connection pool size specified for the MDS Data Source. If you try to export too many portals you might see a `ResourceLimitException` error:

```
Weblogic.common.resourcepool.ResourceLimitException
```

Solution

Export fewer portals. Alternatively, modify the connection pool setting. For details, see the *Oracle Fusion Middleware Performance and Tuning Guide*.

G.8.5 Lists Not Imported Properly

Problem

Lists are not importing properly due to list definition differences in the source and target systems.

Solution

Consider exporting and importing list data. This ensures that list data is consistent with the list definitions being imported.

If you choose to import without data, the list data in the target system is migrated to be consistent with the imported list definitions. If a list column data type is changed, the column values are converted from the target data type to the imported data type, if possible, otherwise the value is deleted. If a list column is removed during import, the column values are deleted.

G.8.6 Exporting and Importing Portals with Tools and Services Configured

Problem

The following error message displays when you try to export a portal with tools and services configured, and try to import the same portal from an instance where some or all of those tools or services are not configured.

```
The following services are not configured: <list of tools and services>. Please
configure these services and try again.
```

Solution

You can work around this problem by either adding the tools and services to the target, or removing the service-related info from the `data.xml` file of the archive as described below.

To remove service-related info:

1. Extract the archive.

   The archive contains two files: `policy-store.xml` and `transport.mar`.

2. Expand the `transport.mar` into a directory.

   The data.xml file is located in the `oracle\webcenter\lifecycle\importexport` directory.

3. Remove the service tags for all the tools and services that are not present in the target as listed in the error message.

   For the example error message above, we would need to remove the following:

```
      <service id="oracle.webcenter.collab.forum" version="11.1.1.0">
         <metadataUsages>
            <metadataUsage includeBaseDocuments="YES"
includeSystemCustomizations="YES">
               <paths>
                  <include
path="/oracle/webcenter/collab/forum/scopedMD/s516227ec_dde1_4991_9e18_28d487cb
3bce/**"/>
               </paths>
            </metadataUsage>
         </metadataUsages>
      </service>
```

```
<service id="oracle.webcenter.collab.rtc" version="11.1.1.0"/>
```

4. Repack the `transport.mar` file by zipping the top-level directories `Oracle` and `pagedefs` into a file named `transport.mar`.
5. Repack the `export` archive by zipping the newly created transport.mar and the `policy-store.xml` file into an archive.
6. Import the new archive.

G.8.7 Tools and Services Disabled After Import

Problem

When you navigate directly to the **Tools and Services** tab in portal administration after importing a portal, all the tools and services are disabled even though they were enabled in the source portal.

This can sometimes happen when the archive you import contains a portal that was exported from a release earlier than WebCenter Portal 11.1.1.8.

Solution

Select the **Enable** check box for tools and services, as required.

Alternatively, open the portal after you import instead of navigating to portal administration. When you access the portal for the first time, tools and services enable automatically.

G.8.8 Importing from the Subportals Page

Problem

When you import a portal from the **Portals** page, the imported portal does not automatically become a subportal of the current portal. The newly imported portal displays in the **Portals** switcher menu, Portals Browser task flow, or the **Portals** page, which display all the portals that are available to you.

Solution

You can import a portal as a subportal by selecting the parent portal on the **Portals** page before you import the archive.

G.8.9 Unable to Import a Portal If the Source and Target Applications Share the Same Content Server

You cannot export/import portals or portal templates between two different WebCenter Portal applications that share the same Content Server.

Similarly, you cannot use the Document Migration Utility to migrate portal documents between two different WebCenter Portal applications that share the same Content Server.

G.8.10 Exporting and Importing Portals in Multibyte Languages

Problem

On Linux, individual portal export or import fails for one or more portals created in multibyte languages due to naming restrictions. Portal names are restricted to alphanumeric and portal characters ("a" through "z", "A" through "Z", "0" through "9", and the single-byte portal character, which WebCenter Portal replaces with "_"(underscore)). If any other characters are used in the portal name, export or import fails.

Solution

Enforce the naming restriction on the server on which Oracle WebCenter Portal is deployed. To do this, set the environment variable `LC_ALL` set to `utf-8`.

Glossary

About mode

A **portlet mode** that typically displays information such as copyright, version, and author of the portlet.

Activity Stream

Feature for viewing the activities tracked for you and other users.

Activity Graph

Provides suggestions of people, portals, and content that a user may be interested in connecting with, based on existing connections and shared interaction with objects in the portal.

The engine used by the Activity Graph tool to provide a central repository for actions that are collected by enterprise applications. The data stored in the activity graph is analyzed to calculate ranks for nodes, predict new actions, and make recommendations.

activity rank

Determines (by the activity graph engine) the relevance of content for search results.

administrator

A person who sets up new machines, administers WebCenter Portal, Portal Framework applications, and databases, and works with other tools such as Fusion Middleware Control and command line tools.

- In WebCenter Portal, administers and sets global options for all portals.
- In Portal Framework applications, manages application-wide settings, assets, users, and roles.

administration console

Enables users with the appropriate privileges to continue developing a Portal Framework application after it has been deployed. Using the administration console, users can also download runtime portal resources (also referred to as assets) and import them back into Oracle JDeveloper for further development. These assets can then be exported from JDeveloper and uploaded back into the deployed application.

Ajax (Asynchronous JavaScript and XML)

An approach using existing standards for exchanging data with a server and updating parts of a web page without reloading the whole page. WebCenter Portal developers use Ajax for UI components, including portlets and pagelets.

Analytics

A WebCenter Portal tools and services offers real-time usage and activity reporting for your portal. In the WebCenter Portal, users can track and analyze WebCenter Portal traffic and usage.

Announcements

Offers a quick, convenient way to create and widely distribute messages instantly or at a specific time.

application customization

Performed by an administrator, all users see the change. These are static changes to an application that affect a site or sites that do not involve changes to the application's code or schema.

See also **user customization** and **personalization**

Application Development Framework (ADF)

A range of technologies aimed at making Java EE application development faster and simpler for developers while at the same time taking advantage of proven software patterns to ensure that the developed application is scalable, performant, and the like.

Application Programming Interface (API)

An application programming interface (API) typically refers to a software library that includes specifications for methods, object classes, data structures, and variables. API also refers to an interface used by software components to communicate with each other. Oracle WebCenter Portal provides a rich assortment of Java, REST, and Expression Language APIs that let you access services and tools, external applications, portal components, data controls, and so on.

application role

Roles that are specific to a particular application and are stored in an application-specific stripe of the policy store.

application skin

Specifies the application background color, screen fonts, and, with some skins, the shapes and images used for application buttons and icons. In WebCenter Portal, the administrator chooses the default application skin and users may change the application skin on the General tab of the Preferences dialog.

application specialist

A WebCenter Portal user who works in Portal Builder to develop site structure by planning, creating, and administering portals and their content. At the application level, an application specialist has the permissions granted to the `Application Specialist` role. In a portal that an application specialist creates, the application specialist can perform the actions available to the `Moderator` role.

application templates (JDeveloper)

Oracle WebCenter Portal provides templates for creating two kinds of applications: Portal Framework applications and Portlet Producer applications. Templates ensure that the right technology scopes are set, tag libraries added, and required Java classes are added to the class path. Once you do this, relevant components are included to the Component Palette and relevant context menus become available in JDeveloper.

See also **Portal Framework application template** and **WebCenter Portal Producer Application template**.

authenticated user

A user who is logged into WebCenter Portal or a Portal Framework application. Credentials of this user are verified against the identity store. By default, an authenticated user can access public information. To access secured information, such as pages and **portlet**s, this user must be authorized through the policy and credential store.

Contrast with **public user**s, who are not logged in, and can access public content only.

authentication

Identification of a user through an identity management system. You can require ADF authentication to enforce credentials for users to access the Portal Framework application only (all ADF resources in the application remain accessible), or authentication *and* authorization to enforce credentials for users to access the Portal Framework application and any ADF resources that have been secured in the application.

authorization

The policies that define the access rights of an individual or group to a secured resource. This resource may be a page or component within a page.

authorized user

An individual who has access to a secured resource. For non-public resources, this individual is also an **authenticated user**.

blog page

A page that provides a personal record of an individual user's experience and opinions. There are two kinds of blog: personal blogs are written by an individual; group blogs are written by several users.

Box layout component

A layout component available through Oracle WebCenter Portal's Composer. A container that enables the placement of content on a page created in the WebCenter Portal. In Composer, a Box is rendered as a rectangle comprised of dashed lines. In a Portal Framework application, this is the runtime equivalent of a Panel Customizable component.

BPEL

Business Process Execution Language. An XML-based markup language for composing a set of discrete web services into an end-to-end process flow.

business role page

A page, created by the WebCenter Portal system administrator, specifically provided for a given role in an organization. Business role pages provide a targeted environment for users of a particular role by delivering information that is timely and relevant to individual roles without the noise of irrelevant information from other lines of business. Business role pages appear in the Home portal of users classified under the specified role.

caching

The act of storing frequently accessed information, typically web pages, in a location where it can be accessed quickly to avoid frequent content generation.

See also **expiry-based caching** and **validation-based caching**.

calendar overlay

The ability to display multiple calendars in a single Events task flow.

Change Mode Button component

In the Composer tag library that enables users to change from page View mode to page Edit mode.

Change Mode Link component

A component provided in the Composer tag library that enables users to change from page View mode to page Edit mode.

check out/check in

A mechanism that enables a user to lock information, by checking it out, so that other users cannot modify that same piece of information. This prevents users from overwriting each other's changes. After making modifications, the user releases it by checking it back in, making it available again for other users to modify.

Child Components

The components contained within a parent component. For example, the task flows contained within a Box layout component are the child components of the Box.

See also **Box layout component** and **parent component**.

chrome

Visual elements surrounding a portlet or task flow that provide an access point for actions, such as those on the Actions menu and those embedded in the chrome itself, such as the minimize icon or resize handles.

CMIS

Content Management Interoperability Services (CMIS) standard defines a domain model and Web services and Restful AtomPub bindings that can be used by applications to work with one or more Content Management repositories or systems.

component

An individual piece of a portal, for example, a task flow, portlet, page, or layout element such as a box or image.

Component Catalog

A dialog, accessed from Composer, that provides access to all the content you can add to a WebCenter Portal application page.

component developer

The developer who builds components (such as portlets, **JavaServer Faces** components, and web services).

Component Properties

A dialog, accessed from Composer, that provides access to a component's parameters, display options, child components, style settings, and associated events.

Connections

Feature for establishing a social network with other portal users (**People Connections**).

Composer

A seamlessly integrated environment for populating, revising, and configuring portal pages. It enables users to easily build or revise page layout and content. It also provides the means of adding different components, such as task flows, portlets, content, and other objects, onto a page and then linking those components for a more relevant or personalized view of the information.

container runtime option

A JSR 286 feature that provides a way to customize the behavior of the portlet container and therefore customize the runtime environment.

content integration tools

Tools provided by **WebCenter Portal tools and services** to enable developers to display content from a **content repository**, such as by creating **data control**s.

Content Presenter

Feature that enables end users to select and search content items and then display those items using available display templates (part of the **Documents**).

content repository

A specialized storage and management mechanism that provides such features as author-based versioning, full text searching, and content categorization and attribution. A content repository is optimized for storing unstructured information, which differentiates it from a data repository.

content repository data control

A **data control** sourced though a content repository. In a **Portal Framework application**, you can create content repository data controls for the following content repositories: Oracle Portal, **Oracle WebCenter Content**, third-party repositories that support the Java Content Repository (JCR) standard, and your local file system.

credential provisioning page

A **JSF** (`*.jspx`) page used for authenticating to an **external application**. At runtime, the Credential Provisioning page displays login data fields consisting of the data fields specified through external application registration. Login information is passed to the producer, which in turn passes the login values to the external application. The application provides the producer with the requested portlets.

After authentication, the user's login credentials are preserved in a **credential store**, which subsequently supplies that information at future sessions. Unless his information changes, the user supplies his credentials only one time.

credential store

Provides storage for login credentials for its associated domain. It also preserves the login credentials that a user provides for authentication to an **external application**. Credential store is usually combined with the policy store as a single logical store.

Although the credentials stored in the credential store are used during subsequent logins for authentication, the main function of this store is to provide authorization for those accounts.

CSS

Cascading Style Sheet. A simple mechanism for ensuring a consistent look and feel or adding style, such as fonts, colors, and spacing, to web documents.

custom action

Icons or menu items rendered on the header or the Actions menu of a Show Detail Frame component surrounding a task flow. Custom actions can represent actions that were defined in the task flow when it was created. For example, at design time a developer can build a task flow with custom personalization settings. At runtime, users can access these settings through icons or Actions menu items provided in the task flow's surrounding chrome (or Show Detail Frame).

custom attribute

In the WebCenter Portal, custom attributes specify information in addition to that provided by the built-in attributes. Custom attributes can be used to determine the content of the components in a portal based on the parameter passed in. For example, a component can display data for a specific customer by passing in the customer ID. A custom attribute is simply a name value pair, for example customerId=400, orderId=11, userName=Smith, and so on. Custom attributes are stored within the portal template.

custom page

Any page created by a user rather than one provided out of the box.

custom display template

A Content Presenter display template is a JSFF file (JSF page fragment) that defines how Content Presenter renders content items on a Portal Framework application page. WebCenter Portal provides several out-of-the-box display templates to get you started, or, you can create your own templates.

custom role

A user role created by an administrator or a portal moderator to meet a specific Home portal or portal requirement.

Customize mode

A **portlet mode** that enables users to set the default values for portlet preferences for all users.

customizable component

A WebCenter Portal component that can be added to a page at runtime to enable end users to perform personalizations such as move, minimize, restore, or remove on content within those components. Customizable components are the **Panel Customizable component** and the **Show Detail Frame component**.

customization

See **application customization**, **personalization**, and **user customization**.

data control

A mechanism that provides an abstraction of the business service's data model. The ADF data controls provide a consistent mechanism for clients and web application controllers to access data objects, collections, methods, and operations.

Data Presenter

Enables you to retrieve data from a data source, such as a relational database or web service, and display that data in your portal as a table, form, or graph.

default language (application-level)

A display language specified by the system administrator that is used when users log in to WebCenter Portal. The administrator sets the application-level default language on the **General** tab of the **Administration** page. Individual users can set their own user-level default language on the General tab of the Preferences dialog.

default language (user preference)

A user-specified display language that is rendered when the user logs in to the WebCenter Portal. This language selection lasts until the user specifies a different default language. It can be overridden by a session language, but returns as the default when the session cookie is purged or expires. This value is set on the **General** tab of the Preferences dialog.

default server

See **Integrated WebLogic Server**.

delegated administration

Provides a mechanism for securing portal resources based on user roles. You apply delegated administration to a page hierarchy, and the specific security assignments are automatically propagated down through the hierarchy through pages and sub pages.

deployment profile

A file used in application deployment that specifies the following types of information:

- The source files, deployment descriptors, and other auxiliary files that are packages
- The type and name of the archive file to be created
- Dependency information
- Platform-specific instructions
- Other information

Design view (Composer)

A view, in **Composer**, that provides a WYSIWYG rendering of the page and its content, where controls are directly selectable on each component. The resource catalog displays inline on the right side of the page, where you can select components to add to the page.

See also **Structure view (Composer)** and **Select view (Composer)**.

device group

Represents a collection of devices that share similar display requirements. Out-of-the-box, WebCenter Portal comes with several pre-configured device groups: Desktop Browsers, iOS Phones, Android Phones, iOS Tablets, and Android Tablets.

discoverable portal

A portal that can be found by anyone logged into WebCenter Portal, for example through a search. Any **Public** or **Private** portal is discoverable. Discoverable portals are listed on the **Portals** page when **All Portals** is selected from the **Show** list. Users wishing to join the portal can request membership through self-service (if enabled) or by contacting the portal moderator.

Discussions

Provides a means of creating and participating in discussion forums.

display language

Controls the language in which application user interface elements, such as buttons, field labels, and screen text, are rendered in the browser. The order of precedence for WebCenter Portal display language settings from weakest to strongest is: browser setting, application setting, portal default language, user preference setting, Change Language task flow, and Global language switcher (public cookie).

Documents page

A system page exposed in portals and the Home portal that provides controls for managing files and folders through the Document Explorer task flow.

Documents

Provides a variety of formats to display folders and files on a page. You can choose the task flows appropriate for your portal to provide features for accessing, adding, and managing folders and files; configuring and viewing file and folder properties; and searching file and folder content in the connected content repositories.

domain

A basic administration unit for WebLogic Server instances. Oracle WebCenter Portal developers using JDeveloper deploy to the Integrated WebLogic Server which is managed within the DefaultDomain. For production purposes, parts of Oracle WebCenter Portal are deployed to Managed Server instances. A domain can have any number of Managed Servers. Managed Servers can be configured to run applications in a test, staging, or production environment, or all three.

dynamically-generated page

A page that displays as the result of a user action, such as a search or a click on a tag. As the name suggests, dynamically-generated pages are not stored, but rather are created as and when needed.

EAR

Enterprise Archive file. A Java EE archive file that is used in deploying applications on a Java EE application server. **Portal Framework application**s are deployed using both a generic EAR file, which contains the application and the respective runtime customization, and a targeted EAR file, which contains only the application for deployment to the application server. EAR files simplify application deployment by reducing the possibility of errors when moving an application from development to test, and test to production.

ECMA-262 specification

A standardization of scripting programming languages, such as **ECMAScript** and JavaScript.

ECMAScript

A scripting programming language, standardized by Ecma International according to the **ECMA-262 specification**. Frequently referred to as JavaScript or JScript, which are both extensions of the ECMA-262 specification.

Edit Defaults mode

(**JSR 286** portlets only.) A **portlet mode** that enables personalization of a JSR 286 portlet. Edit Defaults mode is a display mode for the JSR 286 portlet's properties. In a **Portal Framework application**, the Edit Defaults mode displays on the portlet's Actions menu as the Customize command.

See also **Edit mode**.

Edit mode

A **portlet mode** that enables personalization of the portlet for each user, for each instance.

See also **Edit Defaults mode**.

edit mode

A view mode that enables users to modify the content, style, and layout of a page. See also **Composer**.

EL

Expression Language. Provides a shorthand way of working with web application data by providing operators for retrieving and manipulating application data residing in a Java EE web container. In a **Portal Framework application**, EL expressions are encapsulated in the characters "#{" and "}" and typically come in the form #{object.data} where *object* represents any Java object or **Oracle ADF** component placed in the Java EE web container's page, request, session, or application's scope.

Enterprise Archive file

See **EAR**.

enterprise mashup

An application that enables users to bring all sorts of content and services together in a single place.

Events

Provides calendars for scheduling meetings, appointments, and so on. In WebCenter Portal, it provides calendars to record events relevant to the specific portal. You can also integrate events with Microsoft Exchange Server to enable individual users to access their personal calendars in their Home portal. Personal calendars are also available in Portal Framework applications.

expiry-based caching

A **caching** method that uses a retention period to specify how long the item is valid in the cache before a refresh is required. When there is a request for the item beyond the retention period, it is refreshed in the cache.

See also **validation-based caching**.

Expression Language

See **EL**.

external application

Applications that do not delegate authentication to the single sign-on server. Instead, they display HTML login forms that ask for application user names and passwords. At the first login, users can choose to have the single sign-on server retrieve these credentials for them. Thereafter, they are logged in to these applications transparently.

farm

A collection of components managed by Fusion Middleware Control. A farm can contain a Managed Server domain and other Oracle Fusion Middleware system components that are installed, configured, and running on the domain.

favorites

A personal list of links to favorite pages in WebCenter Portal and external web sites.

Feedback

Feature for posting informal appraisals for and receiving informal appraisals from other portal users (part of **People Connections**).

Portal Framework application

A Portal Framework application is built on top of ADF using the **WebCenter Portals Extension for Oracle JDeveloper**. This application combines web content, portlets, content integration, and collaborative services for the end user. Developers and administrators can create a Portal Framework application based on their roles and skill levels in the organization.

A portal also includes page hierarchies, navigation models, and delegated administration.

Portal Framework application administrator

The administrator responsible for managing and maintaining the **Portal Framework application**. For example, administrators can set application-wide options, manage assets, and grant and revoke privileges.

Also administers Fusion Middleware, deploys Portal Framework applications, and performs on-going administrative tasks for Portal Framework applications and other Oracle WebCenter Portal components through Fusion Middleware Control.

Portal Framework application developer

The developer who plans, builds, and maintains a **Portal Framework application** using the Oracle Application Development Framework, **Oracle JDeveloper**, and **WebCenter Portal tools and services**.

Portal Framework application template

A JDeveloper template which includes WebCenter Portal Framework features like site navigation, page hierarchies, delegated administration, and page templates.

See also **application templates (JDeveloper)** and **WebCenter Portal Producer Application template**.

Full Screen Mode (WebCenter Portal)

A view mode that opens the portal to occupy the entire screen, thus maximizing the display portal. The Sidebar is not displayed in Full Screen Mode.

Full Screen mode (Portlets)

(**PDK-Java** portlets only.) A **portlet mode** that provides more content than can be shown in the portlet when it is sharing a page with other portlets.

Fusion Middleware Control

A browser-based management application that is deployed when you install Oracle WebCenter Portal. From Fusion Middleware Control, you can monitor and administer a **farm** (such as Oracle WebCenter Portal).

Help mode

A **portlet mode** that displays usage information about the functionality of the portlet.

Home portal

A work area within WebCenter Portal that provides individual users with a private portal for storing personal content, keeping notes, viewing and responding to assignments, maintaining a list of online buddies, and performing many other tasks relevant to their unique working day. Users can also extend this environment by creating additional personal pages and custom content.

HTML Markup layout component

A layout component available through Composer. A simple HTML component that renders raw HTML and JavaScript mark-up inline on the page.

Hyperlink layout component

A layout component available through Composer. A link to an internal or external web page. For designers of Portal Framework applications, this is the runtime equivalent of a Go Link component.

Identity Propagation

For a Portal Framework application and associated content repositories, selecting this option allows propagation of current user's identity across the application and processes. The propagated identity is verified on the receiver's side, and then it is used to make decisions such as assigning role based access control.

Image layout component

A layout component available through Composer. An illustration that can include a hyper link. For designers of Portal Framework applications, this is the runtime equivalent of an Image Link component.

IMP

See **Instant Messaging and Presence**.

initialization parameters

The parameters initialized upon the start-up of a standard JSR 286 portlet. Initialization parameters provide an alternative to JNDI (Java Naming and Directory Interface) variables. Use initialization parameters instead of JNDI to configure the behavior of all of the different components of the portlet—for example, servlets and other portlets—in a compatible way. In **WebCenter Portal tools and services**, initialization parameters are entered into the `portlet.xml` file.

Instant Messaging and Presence

Enables users to observe the presence status of other authenticated users and provides instant access to interaction options, such as instant messages and email.

Integrated Development Environment (IDE)

A visual application development tool containing editors, debuggers, screen painters, object browsers, and the like. The Oracle JDeveloper IDE provides a fully featured environment for building custom portal components, like task flows, data controls, managed beans, and scenarios.

Integrated WebLogic Server

A WebLogic Server instance used as a platform for pretesting Portal Framework application deployments on a local computer. Integrated WebLogic Server also contains preconfigured portlet producers and several useful prebuilt portlets.

Iterative development

Iterative development lets you make changes to your Portal Framework application while it is running on the Integrated WebLogic Server and immediately see the effect of those changes simply by refreshing the current page in your browser. The iterative development feature works by disabling certain optimization features. Iterative development allows developers to work more quickly and efficiently when building a Portal Framework application.

Java Content Repository

See **JCR 1.0**.

Java Portlet Specification

Standardizes how components for portal servers are to be developed. This specification defines a common portlet API and infrastructure that provides facilities for personalization, presentation, and security. Portlets using this API and adhering to the specification are product-agnostic, and can be deployed to any portal product that conforms to the specification. See also **JSR 286**.

JavaServer Faces

See **JSF**.

JavaServer Page

See **JSP**.

JCR 1.0

Java Content Repository 1.0. Also known as JSR 170. It proposes a standard access and interaction API for content repositories, much like JDBC does for databases.

JDeveloper

See **Oracle JDeveloper**.

JSF

JavaServer Faces. A standard Java framework for building web applications. It simplifies development by providing a component-centric approach to developing Java web user interfaces. JSF offers rich and robust APIs that provide programming flexibility and ensures that applications are well designed with greater maintainability by integrating the Model-View-Controller (**MVC**) design pattern into its architecture. As JSF is a Java standard developed through Java Community Process, development tools like **Oracle JDeveloper** are fully empowered to provide easy to use, visual, and productive development environments for JSF.

JSF JSP

JavaServer Faces JavaServer Page. JSF JSPs differ from plain JSPs through their support of **Oracle ADF Faces** components for the user interface and JSF technology for page navigation. JSF JSP pages leverage the advantages of the Oracle **Application Development Framework (ADF)** (Oracle ADF) by using the ADF Model binding capabilities for the components in the pages.

JSP

JavaServer Page. An extension to servlet functionality that provides a simple programmatic interface to web pages. JSPs are HTML pages with special tags and embedded Java code that is executed on the web or application server. JSPs provide dynamic functionality to HTML pages. They are actually compiled into servlets when first requested and run in the servlet container.

See also **JSP tags**.

JSP tags

Tags that can be embedded in **JSP**s to enclose Java code. These tags use the `<jsp:` syntax and enclose action elements in the JSP with `begin` and `end` tags similar to XML elements.

JSR 286

Java Specification Request (JSR) 286. Defines a set of APIs for building standards-based portlets using Java. Portlets built to this specification can be rendered to a portal locally or deployed to a WSRP container for rendering portlets remotely. For more information, see `http://jcp.org/en/jsr/detail?id=286`.

JSR 170

See **JCR 1.0**

JSR 329

See **Oracle JSF Portlet Bridge**.

keystore

A file that provides information about available public and private keys that are used for authentication and data integrity. User certificates and the trust points needed to validate the certificates of peers are also stored securely in the keystore.

knowledge worker

A WebCenter Portal user who focuses on providing content and reviewing the content of others. At the application level, a knowledge worker has the permissions granted to the `Authenticated-User` role. At the portal level, a knowledge worker is likely assigned the `Viewer` or `Participant` role.

layout box

A container that enables placement of content on a page created in the WebCenter Portal.

layout component

An object for enhancing the usefulness and appearance of a given page. Layout components include layout boxes, a rich text editor, images, hyperlinks, and so on.

Layout Customizable component

A component provided in the Composer tag library that enables users to select from a set of predefined layouts (for example, two column, three column, two row, and so on) and apply it to the page. Users can apply these layouts to a particular area of the page or to the entire page.

life cycle

The process of creating and testing a portal or Portal Framework application in a design time environment, deploying it to a production system, and then performing

routine maintenance, such as monitoring performance and migrating customization data. The life cycle of portal or a Portal Framework application also includes performing further enhancements, restaging, and then redeploying it to the production system.

Links

Provides a means of creating a bidirectional association between two objects, thus setting up easy access between those objects.

Lists

Provides a means of creating lists and exposing them for placement on portal pages.

Lists page (or console)

A predefined page that displays a portal's current lists.

Mail

Provides a means for exposing familiar email functionality in portals.

Managed Server

In a production environment, a Managed Server hosts applications and the resources needed by those applications. A domain, which is a logically related group of Oracle WebLogic Server resources, can have any number of Managed Servers. An Administration Server manages these servers.

mashup

A web application that enables end users to pull information from different sources to create a customized application that exactly meets their individual requirements.

See also **enterprise mashup**.

MDS

Oracle Metadata Services. A core technology of the **Application Development Framework (ADF)**. MDS provides a unified architecture for defining and using metadata in an extensible and customizable manner.

MDS repository

An application server and Oracle relational database that keep metadata in these areas: a file-based repository, dictionary tables accessed by build-in functions, and a metadata registry. One of the primary uses of MDS is to store customizations and persisted personalization for Oracle applications.

Message Board

Feature for posting messages to and receiving messages from other portal users (part of **People Connections**).

metadata

Information about a content item, such as title, author, or security group. Metadata is used to describe, find, and group content items. Also referred to as content information.

Model-View-Controller

See **MVC**.

moderator

A WebCenter Portal user who is responsible for managing a particular portal. A portal moderator can add and remove members, invite new members, enable self registration, provide and update portal metadata, and manage the tools and services available to the portal.

Movable Box layout component

A layout component available through Composer. A container that enables the placement of content on a page created in the WebCenter Portal. Movable Boxes, along with their content, can be moved around on the page. For designers of Portal Framework applications, this is the runtime equivalent of Show Detail Frame component.

MVC

Model-View-Controller. A classic design pattern often used by applications that need the ability to maintain multiple views of the same data. The MVC pattern hinges on a clean separation of objects into one of three categories: models for maintaining data, views for displaying all or a portion of the data, and controllers for handling events that affect the model or views. Because of this separation, multiple views and controllers can interface with the same model. Even new types of views and controllers that never existed before, such as portlets, can interface with a model without forcing a change in the model design.

navigation

WebCenter Portal provides three navigation components to create portal navigation. These components are: Breadcrumb navigation, menu navigation, and tree navigation.

navigation model

Navigation models provide data to the navigation user interface and enable navigation to assets in your portal, such as pages, page hierarchies, task flows, external sites, portlets, and other entities.

Notes

Provides useful features for writing personal notes and reminders.

Notifications

Provides an automated means of triggering notices across different messaging channels, such as phone, mail, worklist, and so on. Messages are triggered when the portals and application objects to which you have subscribed change.

OmniPortlet

A component of **WebCenter Portal tools and services** that enables you to inject portal-like capabilities, such as portlets, content integration, and customization, into your **Oracle ADF Faces** applications.

Oracle ADF

Oracle Application Development Framework. A range of technologies aimed at making Java EE application development faster and simpler for developers while at the same time taking advantage of proven software patterns to ensure that the developed application is scalable, performant, and the like.

Oracle Access Manager (OAM)

Part of Oracle's enterprise class suite of products for identity management and security, Oracle Access Manager provides a wide range of identity administration and security functions, including several single sign-on options for WebCenter Portal and Portal Framework applications. OAM is the recommended single sign-on solution for Oracle WebCenter Portal 11g installations.

Oracle ADF Faces

Oracle ADF Faces is a rich set of user interface components based on the new **JavaServer Faces** JSR (JSR 127). Oracle ADF Faces provide various user interface components with built-in functionality, such as data tables, hierarchical tables, and color and date pickers, that can be customized and reused in an application.

Oracle Enterprise Manager

A component that enables administrators to manage Oracle Fusion Middleware services through a single environment. The administrator uses Enterprise Manager to configure, manage, and monitor Portal Framework applications.

Oracle HTTP Server (OHS)

Software that processes web transactions that use the Hypertext Transfer Protocol (HTTP). Oracle uses HTTP software developed by the Apache Group.

Oracle Internet Directory

Oracle's LDAP V3 compliant LDAP server. It is used as a repository for provisioning users and groups. By default, the **Oracle Single Sign-On (OSSO)** authenticates user credentials against Oracle Internet Directory information about dispersed users and network resources. Oracle Internet Directory combines LDAP version 3 with the high performance, scalability, robustness, and availability of the Oracle database.

Oracle JDeveloper

Oracle JDeveloper is an integrated development environment for building applications and web services using the latest industry standards for Java, XML, and SQL. Developers can use Oracle JDeveloper to create Java portlets, Portal Framework applications, portlets, skins, portal templates, task flows, mBeans, data controls, and so on.

Oracle JSF Portlet Bridge

Based on and conforming to JSR 329, the Oracle JSF Portlet Bridge enables application developers to expose a JSF application or task flow as a JSR 286 portlet for consumption in another application.

Oracle Metadata Services

See **MDS**.

Oracle Secure Enterprise Search (SES)

Provides easy-to-use search for public and secure data, with unified ranking results. With Portal Framework applications, Oracle SES is set as the default and preferred search platform.

With WebCenter Portals, WebCenter Portal's internal live search adapters are set as the default search platform; however, large-scale implementations should be configured to use Oracle SES for best performance.

Oracle Single Sign-On (OSSO)

A component that enables users to log in to all features of the Oracle Fusion Middleware product suite, and to other web applications, using a single user name and password.

Oracle Technology Network

See **OTN**.

Oracle WebCenter Content

Provides a flexible, secure, centralized, web-based repository that manages all phases of the content life cycle: from creation and approval to publishing, searching, expiration, and archival or disposition. It enables contributors to easily contribute content from native desktop applications, efficiently manage business content through rich library services, and securely access that content anywhere using a web browser. All content, regardless of content type, is stored in the web repository or database for management, reuse and access.

Oracle WebCenter Content Server

A content repository for building secure business libraries with checkin and checkout, revision control, and automated publishing in web-ready formats. Current information is available to authorized users anytime, anywhere.

Oracle WebCenter Content Server is a component of Oracle WebCenter Content.

Oracle WebCenter Content's Site Studio

A powerful, flexible web development application suite that offers a comprehensive approach to designing, building, and maintaining enterprise-scale web sites. Site Studio uses Oracle WebCenter Content: Content Server as the main repository for a web site.

In WebCenter, Content Presenter integrates with Site Studio to allow you to create, access, edit, and display Site Studio contributor data files in either a Site Studio region template or a custom Content Presenter display template.

Oracle WebCenter Portal's Discussions Server

Backend discussions server for discussions and announcements.

Oracle WebCenter Portal Framework

An WebCenter Portal Framework application is a standard ADF web application that includes portal features, like navigation, pages, page templates, content integration, and so on.

Oracle WebCenter Portal's Pagelet Producer

Provides a collection of useful tools and features that facilitate dynamic pagelet development.

Oracle WebLogic Server Administration Console

A browser-based environment for managing WebLogic Server, including deployed applications, domains, security, clusters, and so on.

OTN

Oracle Technology Network. The online Oracle technical community that provides a variety of technical resources for building Oracle-based applications. You can access OTN at `http://www.oracle.com/technetwork`.

Page Customizable component

A component provided in the Composer tag library that defines the editable area of a page at runtime. Within this area, users can edit properties for a component, add content to the page, arrange content, and so on.

page hierarchy

A model, in Portal Framework applications, that associates pages in a parent-child relationship, where any page can have one or more sub pages. This parent-child model not only helps define the overall structure of the portal, but also allows child pages to inherit the security policies from their parent.

page parameter

A parameter associated with a page that can be used to store values that can then be passed to the components on the page. It also enables your page to take values through its URL. Page parameters are defined using the `<parameter>` tag at the top of your `PageDef.xml`. You can bind page parameters to your **page variable**s.

Page Properties

A dialog, accessed from Composer, that provides access to a page's display options, security settings, and parameters.

page scheme

Determines the background image used in the page. The WebCenter Portal provides several default page schemes and an option for specifying a custom page scheme.

page style

Determines the initial page structure, for example one column or two column. Some default page styles also include the task flows, components, and page properties useful for a particular purpose. For example, a page created using the Text page style includes a Text layout component.

page template

Lets you specify view elements that you intend to be common to all of your pages. A page template file is a JSPX file that includes ADF layout components and other elements. Typically, page templates define a page layout, with headers, footers, and content areas. In addition, the page template usually specifies the positioning and style of the navigation UI for your pages.

page variable

A variable that binds your public portlet parameter to the page. Page variables are defined within the `<variableIterator>` of your `PageDef.xml`. One page variable can be bound to multiple public portlet parameters.

Panel Customizable component

A component provided in the Composer tag library that provides a container region for a group of Oracle ADF components and portlets that are customizable at runtime. Any Show Detail Frame components and portlets added as child components to a Panel Customizable component can be moved or maximized with the Panel Customizable component.

parent component

A component that contains other components, such as a Box layout component that contains task flows. The Box is the parent component of the task flows. In contrast, the task flows are the Box's child components.

See also **Child Components**.

participant

A WebCenter Portal user who can manipulate the content of a portal. A participant can upload and share documents, initiate and take part in chats with other members, create discussion topics, create new or view existing lists.

PDK-Java

Java Portlet Developer Kit. The development framework used to build and integrate web content and applications with **WebCenter Portal tools and services**. It includes toolkits, samples, and technical articles that help make portal development simple. You can take existing Java **servlet**s, **JSP**s, URL-accessible content and web services and turn them into **portlet**s. It is typically used by external developers and vendors to create portlets and services.

People Connections

Provides social networking tools for creating, interacting with, and tracking the activities of one's enterprise connections.

See also, **Activity Stream**, **Connections**, **Feedback**, **Message Board**, and **Profile**.

personalization

Dynamic changes to an application's behavior based on user context, facilitated by Personalization for WebCenter Portal.

See also **application customization** and **user customization**

Personalization for WebCenter Portal

Enables you to deliver content within your application to targeted users based on selected criteria. Personalization for WebCenter Portal also provides a declarative means for specifying dynamic application flow.

personal page

A page created by a user in his or her Home portal. Personal pages are viewable by other users only if specifically granted access by the user who created the page.

personal profile

A page that displays a user's personal information such as email address, phone number, office location, department, manager, direct reports, and so on.

See also, **Profile**.

Polls

Enables you to create, edit, and take online polls on your portal pages. Polls let you survey your audience (such as their opinions and their experience level) and check whether they can recall important information, and gather feedback on the efficacy of presentations.

portal

A common interface (that is, a web page) that provides a personalized, single point of interaction with web-based applications and information relevant to individual users or class of users.

Portal Builder

Comprises the portal creating, editing, and administration areas of WebCenter Portal. In Portal Builder, you can create a portal, add and edit the pages of a portal in the page editor (Composer), and administer a single portal as the portal owner. The system administrator has access to the Portal Builder administration area that allows for administering all portals.

Portals page

A predefined page in Portal Builder that displays a list of all the portals available to the currently logged in user. The user can choose to display **All Portals**, only portals of which the user is a member (**Joined**), only portals of which the user is the moderator (**Moderated**), portals that are accessible to all (**Public**), and portals that have been made discoverable (**Discoverable**).

portal application template

See **application templates (JDeveloper)**.

Portal Developer Kit

See **PDK-Java**.

portlet

A reusable web component that can draw content from many different sources. Portlets can display excerpts of other web sites, generate summaries of key information, perform searches, and access assembled collections of information from a variety of data sources. Because different portlets can be placed on a common page, the user receives a single-source experience, even though the content may be derived from multiple sources. Portlet resources include the many prebuilt portlets available out of the box and programmatic portlets built through WebCenter Portal's JSR 286, PDK-Java Portlet wizards, and other portlet building tools.

portlet event

A JSR 286 feature that allows portlets to react to actions or state changes not directly related to an interaction of the user with the portlet.

portlet filter

A JSR 286 feature that allows on-the-fly transformations of information in both the request to and the responses from a portlet. A portlet filter is a reusable piece of code that can transform the content of portlet requests and portlet responses.

portlet mode

The ways by which a **portlet** can be called to display information. These methods include:

- **Shared Screen mode** or **View mode**
- **Edit mode** or **Edit Defaults mode**
- **Customize mode**
- **Help mode**

- **About mode**
- **Full Screen mode (Portlets)** or **Show Details Page mode**

predefined page

A page created by the WebCenter Portal to perform a specific function. Examples of predefined pages include, Welcome pages, Search pages, and Documents pages.

predeployment tool

A utility for **Portal Framework application**s that assists you in configuring your target system with the new producer registrations you have added to your application in Oracle JDeveloper. You must run this utility before deploying your application. You can also use this utility after deployment to migrate metadata from stage to production, for example, to export and import your customizations. This tool also enables you to define the **MDS** repository location to allow run-time customizations to be migrated.

pretty URL

A shortened version of a page's URL that hides the complexity of the real web address.

private parameter

A portlet parameter that is known only to the portlet itself and has no connection to the page on which the portlet resides.

Contrast with **public parameter**.

producer

A communication link between portlet consumers (such as a **Portal Framework application** or a **portal**). When a consumer application renders a portlet, it calls the producer of that portlet, which in turn executes the portlet and returns the results in the form of portlet content. A producer can contain one or more portlets. A portlet can be contained by only one producer.

WebCenter Portal tools and services supports two types of producers:

- Oracle **PDK-Java** producers: Deployed to a Java EE application server, which is often remote and communicates through Simple Object Access Protocol (SOAP) over HTTP.
- **Web Services for Remote Portlets** (WSRP): A web services standard that enables the plug-and-play of visual, user-facing web services with portals or other intermediary web applications. Being a standard, WSRP enables interoperability between a standards-enabled container based on a particular language (such as **JSR 286**, .NET, Perl) and any WSRP portal. A portlet (regardless of language) deployed to a WSRP-enabled container can be rendered in any application that supports this standard.

Profile

Feature for viewing and managing information about yourself, such as your contact information, manager, and direct reports, and for viewing this information about other application users (part of **People Connections**).

programmatic portlets

Portlets constructed in a non-declarative manner using APIs. Also referred to as *hand-* or *manually-coded* portlets.

public render parameters

A JSR 286 feature that enables portlets to share parameter values, allowing a form of interportlet communication.

public portal

A portal that is available to all users, even those who are not logged in to the WebCenter Portal.

public page

A page within the WebCenter Portal that is available to all users, even those who are not logged in to the application.

public parameter

A portlet parameter that is known to the page and bound to it by way of a page variable.

Contrast with **private parameter**.

public user

A user who can access, but is not logged into a portal or Framework application. A public user can view any page that has been marked as public, but cannot personalize or edit any content, or view pages that have any form of access control.

Contrast with **authenticated user**.

Recent Activities

Provides a means of tracking recent activities in a Portal Framework application.

recipe

A weighted list of similarity calculations. The weighting of each calculation determines its significance in deciding the overall recommendation score. Recommendations are ordered by their total recommendation score.

resize handle

A user interface element in a task flow chrome increasing or decreasing the height of the task flow.

asset

An object that defines the structure, look and feel, or content of a portal. Assets include page templates, navigation models, resource catalogs, skins, page styles, Content Presenter display templates, task flow styles, pagelets, task flows, and data controls.

Resource Action Handling framework

Enables services that expose custom resources to be viewed, searched, and tagged.

resource catalog

A catalog that provides a federated view of one or more otherwise unrelated repositories in a unified search and browse user interface. Resources are created and published in their source repository and are then exposed to the developer in JDeveloper's Resource Palette and to the end user in the resource catalog viewer. Resource catalogs can contain layout components, Oracle ADF components, portlets, task flows, and documents.

Resource Index

The starting point for accessing WebCenter Portal REST APIs. Sending a GET request to the Resource Index URI returns a list of links to entry points for all available services.

resource type

Defines the type of resource that a WebCenter Portal REST API link identifies. Use resource types to determine the response bodies for GET requests and allowable request bodies for POST and PUT. Also use `resourceType` attributes on entities to uniquely identify their type.

REST APIs

Oracle WebCenter Portal provides a set of web-based REST (REpresentational State Transfer) APIs for retrieving and modifying server data dynamically from the client. REST APIs are available for many WebCenter Portal tools and services.

Reverse Proxy Server

A server process that hides the physical location of internal servers by exposing the servers as a single public site. Requests to the public site are routed to the appropriate internal server.

round-trip development

Round-trip development refers to features and techniques that allow you to retrieve resources from a deployed, runtime portal back to JDeveloper for maintenance or enhancement. After modifying a resource in JDeveloper, you can use the administration console to upload the resource back to the deployed portal. WebCenter Portal's round-trip development features provide a simple, convenient way to modify portal resources without redeploying the entire application.

RSS reader

An RSS reader provided with the WebCenter Portal that incorporates public news feeds from external sources onto portal pages. This RSS reader is available only in WebCenter Portal, and not in Portal Framework applications.

RSS

Provides a means of publishing content from other services as news feeds. The RSS tool supports both RSS 2.0 and Atom 1.0 formats.

Search

Enables the discovery of information and people in a portal, returning only the results users are authorized to see.

Secure Enterprise Search

See **Oracle Secure Enterprise Search (SES)**.

Select view (Composer)

A view, in Composer, that provides a WYSIWYG rendering of the page and its content, where a component can be selected for quick access to its properties or the properties of its parent component. Component cannot be deleted in Select view.

See also **Structure view (Composer)** and **Design view (Composer)**.

Self-Registration page

A predefined page where users can register with the WebCenter Portal, thus creating themselves an identity store login account. Administrators can customize certain aspects of this page.

Self-Subscription page

A predefined page where users can register to become members of a portal. Moderators can customize certain aspects of this page.

service ID

In Expression Language, the string that identifies a particular service. For example, the string `oracle.webcenter.collab.announcement` is the service ID for the Announcements service.

A PDK-Java producer's unique identifier. PDK-Java enables you to deploy multiple producers under a single adapter servlet. Different producers are identified by their unique service IDs. A service ID is required only when a service ID/producer name is not appended to the URL endpoint.

servlet

A Java program that usually runs on a Web server, extending the web server's functionality. HTTP servlets take client HTTP requests, generate dynamic content (such as through querying a database), and provide an HTTP response.

session language

A display language specified by the user that remains in effect for the life of the session cookie (from log on to log off). If the user clears browser cookies, the display language reverts to the user-level default language, if specified, then to the application-level default language set by the administrator. Set the session language in the Change Language pop-up, accessible from the Welcome page.

Shared Screen mode

A **portlet mode** that renders the body of the portlet and enables you to display a portlet on a page that can contain other portlets. Every portlet must have at least a Shared Screen mode.

See also **View mode**.

Show Detail Frame component

A component provided in the Composer tag library that renders a border or chrome around the child component. It provides a header with an Actions menu and thereby provides user interface (UI) controls to customize the display of the child component. However, to customize the display of the child component, the Show Detail Frame component must be included inside a Panel Customizable component.

Show Details Page mode

A **portlet mode** that provides full-browser display of the portlet. For example, a portlet in **Show Page mode** could be limited to displaying only the ten most recently submitted expense reports, while the same portlet in Show Details Page mode could show all submissions.

Contrast with **Show Page mode**.

show modes

Types of **portlet mode**s encompassing **Show Page mode** and **Show Details Page mode**.

Show Page mode

A **portlet mode** that provides a smaller portlet display to allow portal for additional portlets and other objects in the browser window. For example, a portlet in Show Page mode could be limited to displaying only the ten most recently submitted expense reports, while the same portlet in Show Details Page mode could show all submissions.

Contrast with **Show Details Page mode**.

similarity calculation

Used by the activity graph to provide a similarity score (a number between zero and one) that designates how similar two objects are to each other given a specific criterion. The weighting of each calculation determines its significance in deciding the overall recommendation score. Recommendations are ordered by their total recommendation score.

skin

A style sheet based on the CSS 3.0 syntax specified in one place for an entire portal. Instead of providing a style sheet for each component, or inserting a style sheet on each page, you can create one skin for the entire portal.

Structure view (Composer)

A view, in Composer, that provides a combined WYSIWYG and hierarchical rendering of page components, where controls are available in the header of the hierarchical list pane to add, edit, delete, hide, and rearrange page components.

See also **Design view (Composer)** and **Select view (Composer)**.

portal

A work area within WebCenter Portal that supports a group of people of any size that is organized around an area of interest or a common goal.

portal icon

An image displayed alongside portal names on the Portals page in My Portals to help other users with identification and location.

portal logo

An image displayed on the Home portal page to provide a visual identity for the portal. The Home portal logos also display alongside the portal name at the top of the page in Full Screen Mode.

portal member

A user who is participating in a portal. Members can be added or invited to a portal, or they can subscribe to a portal themselves if self-registration is enabled.

portal owner

A user who initially created a portal. The portal owner is automatically also a moderator of the portal.

portal template

A starting point for creating a new portal. WebCenter Portal includes several out-of-the-box templates to get you started, and you can create custom portal templates using existing portals as the basis.

Portal Unavailable page

A predefined page that displays when a portal member tries to open a portal that is temporarily offline. Moderators can customize this page.

Portals Switcher

Provides access to a popup window where users can select a portal to which to navigate. **Recent Portals** lists up to ten recently accessed portals, followed by portals to which current user most recently gained access. **Portals** lists all portals to which the current user has access, in alphabetical order. A list of links provides direct access from the menu to the Home portal, the portals browser page, the Create a Portal dialog, and Portal Builder.

Tags

Enables users to apply their own terms to portal objects, making it possible to search for those objects using personally meaningful terms.

task flow

A set of ADF Controller activities, control flow rules, and managed beans that interact to allow a user to complete a task. Task flows provide a modular approach for defining control flow in a portal. Instead of representing a portal as a single JSF page flow, developers can break it up into a collection of reusable task flows.

template

See **portal template**, **application templates (JDeveloper)**, **Portal Framework application template**, **WebCenter Portal Producer Application template**, **page template**, and **custom display template**.

Text layout component

A layout component available through Composer. A rich text editor for providing static page text. For designers of Portal Framework applications, this is the runtime equivalent of a Rich Text Editor component.

Unauthorized Access page

A predefined page that is shown when someone without access permission tries to open a page.

URL parameter

See **private parameter**.

user customization

Changes that affect only a user's own work space.

See also **application customization** and **personalization**

validation-based caching

A **caching** method that uses a validation check to determine if the cached item is still valid.

Contrast with **expiry-based caching**.

Virtual Content Repository

Virtual Content Repository (VCR) enables you to plug in multiple, heterogeneous content repositories.

View mode

(**JSR 286** portlets only.) A **portlet mode** that enables you to display a JSR 286 portlet on a page that can contain other portlets. It is the only required mode for JSR 286 portlets.

See also **Shared Screen mode**.

Web 2.0

Technologies, such as wiki, RSS, and blogs, that enable the construction of highly interactive web applications.

See also **WebCenter Portal tools and services**.

WebCenter Portal

Out-of-the-box application built using JSF, Oracle ADF, WebCenter Portal Framework, WebCenter Portal tools and services, and Composer. WebCenter Portal provides a browser-based platform for creating enterprise portals, multiple sites and communities, a Home portal for each user, and threaded discussions, blogs, wikis, worklists, announcements, RSS, recent activities, search, and more.

WebCenter Portal Producer Application template

An application template, provided by JDeveloper, for creating an application with the recommended projects and technology scopes required for developing portlets. The Portlet Producer Application template consists of a single project scoped for portlet creation (Portlets).

See also **Portal Framework application template** and **producer**.

Web Page layout component

A layout component available through Composer. A means of embedding another web site, wiki, or blog within the context of a page which is created in the WebCenter Portal. For designers of Portal Framework applications, this is the equivalent of an Inline Frame component.

Web Services for Remote Portlets

See **WSRP**.

WebCenter Portals Extension for Oracle JDeveloper

An extension available through the Oracle JDeveloper Update Wizard that installs the necessary libraries, templates, wizards, and dialogs needed to build and deploy **Portal Framework application**s in **Oracle JDeveloper**.

WebCenter Portal Framework

See **Oracle WebCenter Portal Framework**.

WebCenter Portal systems administrator

See administrator.

WebCenter Portal tools and services

A collection of tools and services that expose social networking and personal productivity features.

- Activity Graph
- Announcements
- Analytics
- Discussions
- Documents
- Events
- Instant Messaging and Presence
- Links
- Lists
- Mail
- Notes
- Notifications
- People Connections
- Personalization for WebCenter Portal
- Polls
- Recent Activities
- RSS
- Search
- Tags
- Worklist

Welcome page

There are two types of Welcome page:

- Public Welcome page: A predefined page that users encounter before logging in to the WebCenter Portal.
- Personal Welcome page: A predefined page that introduces users to their Home portal.

WebLogic Server (WLS)

WebLogic Server. A scalable, enterprise-ready Java Platform, Enterprise Edition (Java EE) application server. The WebLogic Server infrastructure supports the deployment of many types of distributed applications and is an ideal foundation for building applications based on Service Oriented Architectures (SOA).

See also **Integrated WebLogic Server**

WLST

WebLogic Scripting Tool. A command line tool for managing Oracle Fusion Middleware components, such as Oracle WebCenter Portal.

Worklist

Provides access to notifications, alerts, and BPEL tasks assigned to the current user.

WSRP

Web Services for Remote Portlets (WSRP) is a web services standard that allows the plug-and-play of visual, user-facing web services with portals or other intermediary web applications. Being a standard, WSRP enables interoperability between a standards-enabled container based on a particular language (such as JSR 286, .NET, Perl) and any WSRP portal. A portlet (regardless of language) deployed to a WSRP-enabled container can be rendered on any portal that supports this standard.