Oracle® Fusion Middleware

Administering Oracle WebCenter Portal

12*c* (12.2.1.4.0)
E95538-12
April 2023

ORACLE

Oracle Fusion Middleware Administering Oracle WebCenter Portal, 12*c* (12.2.1.4.0)

E95538-12

Copyright © 2007, 2023, Oracle and/or its affiliates.

Primary Author: Oracle Corporation

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle®, Java, and MySQL are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

Preface

Audience xxxv
Documentation Accessibility xxxv
Diversity and Inclusion xxxv
Related Documents xxxvi
Conventions xxxvi

Who's Who

Knowledge Worker xxxvii
Application Specialist xxxviii
Web Developer xxxix
Developer xl
System Administrator xl

Part I Introduction to Oracle WebCenter Portal

1 Introduction to Administration for WebCenter Portal

Introducing Oracle WebCenter Portal 1-1
Oracle WebCenter Portal Architecture 1-2
- WebCenter Portlets 1-2
- Application Development Framework 1-2
- Portal Composer 1-2
- Tools and Services 1-3
- Discussion Server 1-3
- Analytics 1-4

Oracle WebCenter Portal Topology 1-4
- Oracle WebCenter Portal Directory Structure 1-4
- Oracle WebCenter Portal Managed Servers 1-6
- Oracle WebCenter Portal Configuration Considerations 1-6
- Discussions Server Configuration 1-7

Oracle WebCenter Portal State and Configuration Persistence 1-7
Analytics Considerations 1-7
Oracle WebCenter Portal Log File Locations 1-8
Understanding the Oracle WebCenter Portal Installation 1-8
Understanding Administrative Operations, Roles, and Tools 1-8
Performance Monitoring and Diagnostics 1-10
Understanding Security 1-10
Data Migration, Backup, and Recovery 1-10
Oracle WebCenter Portal Administration Tools 1-11
Oracle Enterprise Manager Fusion Middleware Control Console 1-11
Displaying Fusion Middleware Control Console 1-11
Oracle WebLogic Server Administration Console 1-12
Locking Domain Configuration 1-12
Oracle WebLogic Scripting Tool (WLST) 1-13
Running Oracle WebLogic Scripting Tool (WLST) Commands 1-13
System MBean Browser 1-15
WebCenter Portal Administration Pages 1-16

Part II Getting Started

2 Getting Started Administering WebCenter Portal

Role of the System Administrator 2-2
Installing WebCenter Portal 2-2
Setting Up WebCenter Portal for the First Time (Roadmap) 2-2
Customizing WebCenter Portal for the First Time (Roadmap) 2-5
System Administration for WebCenter Portal – Fusion Middleware Admin Role (Roadmap) 2-7
System Administration for WebCenter Portal – WebCenter Portal Admin Role (Roadmap) 2-9

3 Starting Enterprise Manager Fusion Middleware Control

Displaying Fusion Middleware Control Console 3-1
Navigating to the Home Page for WebCenter Portal 3-2
Home Page for WebCenter Portal 3-2
Navigating to the WebCenter Portal Home Page 3-5
Navigating to Dependent Components 3-6

4 Starting and Stopping Managed Servers and Applications for Oracle WebCenter Portal

Starting Node Manager 4-2

Starting and Stopping Managed Servers for WebCenter Portal Application Deployments 4-2
Oracle WebCenter Portal Managed Servers 4-2
Starting and Stopping Managed Servers 4-3
Starting and Stopping the WebCenter Portal Application 4-4
Starting WebCenter Portal Using Fusion Middleware Control 4-5
Starting WebCenter Portal Using WLST 4-5
Stopping WebCenter Portal Using Fusion Middleware Control 4-5
Stopping WebCenter Portal Using WLST 4-6

Part III Administering Tools and Services

5 Managing Tools and Services

About Tools and Services 5-1
Enabling and Disabling Tools and Services in WebCenter Portal 5-7

6 Managing Connections to Oracle WebCenter Content Server

About Oracle WebCenter Content Server Connections 6-1
Prerequisites for Configuring Oracle WebCenter Content Server 6-3
Installation Prerequisites for Oracle WebCenter Content Server 6-3
Installation Prerequisites for Inbound Refinery 6-4
Configuration Prerequisites for Oracle WebCenter Content Server and Inbound Refinery 6-4
Security Prerequisites for Oracle WebCenter Content Server and Inbound Refinery 6-5
Configuration Roadmap for Oracle WebCenter Content Server 6-5
Creating a Connection to Oracle WebCenter Content Server 6-8
About Creating a Connection to Oracle WebCenter Content Server 6-8
Creating a Connection to Oracle WebCenter Content Server Using Fusion Middleware Control 6-10
Connecting to Oracle WebCenter Content Server Using Socket-Based Communication 6-11
Connecting to Oracle WebCenter Content Server Using Secure Socket-Based Communication 6-14
Connecting to Oracle WebCenter Content Server Using JAX-WS 6-18
Connecting to Oracle WebCenter Content Server Using HTTP 6-20
Registering Oracle WebCenter Content Server Using WLST 6-22
Oracle WebCenter Content Server Connection Parameters for RIDC Socket Types 6-23
Configuring Oracle WebCenter Content Server 6-25
Creating a Custom Administrator 6-26
Enabling Mandatory Components 6-27
Enabling the FrameworkFolders Component 6-28

Enabling the WebCenterConfigure Component 6-28
Configuring the Inbound Refinery 6-30
Creating an Outbound Provider 6-30
Selecting the File Formats To Be Converted 6-31
Enabling the Conversion of Wikis and Blogs into PDFs 6-32
Specifying the Timeout Setting for File Conversions 6-33
Setting Up SSL for Oracle WebCenter Content Server 6-34
Setting Up Site Studio 6-34
Enabling the iFraming UI 6-35
Enabling Full-Text Search 6-36
Creating Content Profiles in Oracle WebCenter Content Server 6-36
Configuring Access Control List Security 6-37
Enabling Digital Asset Manager 6-38
Additional Optional Configurations for Oracle WebCenter Content Server 6-38
Configuring Oracle WebCenter Content Server for Desktop 6-39
Configuring the File Store Provider 6-40
Setting Up Node Manager 6-41
Configuring Localization Properties 6-41
Showing and Hiding the Wiki Markup Tab in the Rich Text Editor 6-41
Disabling Text Wrapping in the Rich Text Editor 6-42
Registering the Default Oracle WebCenter Content Server Repository 6-43
Configuring the Default Oracle WebCenter Content Server Connection for Oracle WebCenter Portal 6-43
Checking the Oracle WebCenter Portal Data Seeded in Oracle WebCenter Content Server 6-43
Setting Connection Properties for the Default Oracle WebCenter Content Server Connection 6-46
Setting Connection Properties for the Default Oracle WebCenter Content Server Connection Using Fusion Middleware Control 6-46
Setting Connection Properties for the Default Oracle WebCenter Content Server Connection Using WLST 6-48
Modifying Oracle WebCenter Content Server Connection Details 6-49
Modifying Oracle WebCenter Content Server Connection Details Using Fusion Middleware Control 6-49
Modifying Oracle WebCenter Content Server Connection Details Using WLST 6-49
Modifying Cache Settings for Content Presenter 6-50
Configuring the Cache to Check for External Oracle WebCenter Content Server Changes 6-57
Modifying Oracle WebCenter Content Server's Contributor Data Files 6-57
Modifying Oracle WebCenter Content Server's Cache Invalidation Interval 6-58
Testing the Cache Settings 6-59
Deleting Oracle WebCenter Content Server Connections 6-62

Deleting Oracle WebCenter Content Server Connections Using Fusion Middleware Control 6-63
Deleting Oracle WebCenter Content Server Connections Using WLST 6-63
Changing the Maximum File Upload Size 6-63
Configuring Content Manager for Oracle Content and Experience Cloud 6-64

7 Managing Analytics

About Analytics in WebCenter Portal 7-2
Analytics Components 7-2
Analytics Task Flows 7-3
Configuration Roadmap for Analytics 7-4
Analytics Prerequisites 7-5
Analytics – Installation 7-5
Analytics – Configuration 7-5
Analytics – Security Considerations 7-6
Analytics – Limitations 7-6
Configuring Analytics Collector Settings 7-6
Setting Analytics Collector Properties Using WLST 7-7
Setting Analytics Collector Properties Using Fusion Middleware Control 7-7
Registering an Analytics Collector for Your Application 7-9
Registering an Analytics Collector Using Fusion Middleware Control 7-9
Registering an Analytics Collector Using WLST 7-11
Disabling WebCenter Portal Event Collection 7-11
Disabling WebCenter Portal Event Collection Using Fusion Middleware Control 7-11
Disabling WebCenter Portal Event Collection Using WLST 7-12
Validating Analytic Event Collection 7-12
Viewing the Current WebCenter Portal's Analytic Event List 7-13
Purging Analytics Data 7-14
Partitioning Analytics Data 7-14

8 Managing Calendar Events

About Events Connections 8-1
Configuring Personal Events for WebCenter Portal 8-2
Events Prerequisites for Personal Events 8-3
Microsoft Exchange Server 2013 Prerequisites 8-3
Microsoft Exchange Server 2013 - Installation 8-3
Microsoft Exchange Server 2013 - Configuration 8-4
Microsoft Exchange Server 2013 - Security Considerations 8-4
Microsoft Exchange Server 2013 - Limitations 8-6
Microsoft Exchange Server 2010 Prerequisites 8-6

Microsoft Exchange Server 2010 - Installation 8-7
Microsoft Exchange Server 2010 - Configuration 8-7
Microsoft Exchange Server 2010 - Security Considerations 8-7
Microsoft Exchange Server 2010 - Limitations 8-9
Registering Events Servers 8-9
Registering Events Servers Using Fusion Middleware Control 8-9
Registering Event Servers Using WLST 8-10
Choosing the Active Events Server Connection 8-11
Choosing the Active Events Server Using Fusion Middleware Control 8-11
Choosing the Active Events Server Connection Using WLST 8-11
Modifying Events Server Connection Details 8-12
Modifying Events Server Connection Details Using Fusion Middleware Control 8-12
Modifying Events Server Connection Details Using WLST 8-12
Deleting Event Server Connections 8-13
Deleting Event Server Connections Using Fusion Middleware Control 8-13
Deleting Event Server Connections Using WLST 8-13

9 Integrating Other Oracle Applications

About Integrating Other Oracle Applications 9-1
Integrating Siebel Applications 9-2
How to Integrate Siebel Applications as Web Services 9-3
How to Prepare the Siebel Application 9-3
How to Consume a Siebel Web Service Data Control 9-5
Integrating E-Business Suite Applications 9-6
About Integrating EBS Applications 9-7
Understanding EBS Integration 9-7
Requirements for Integrating EBS Applications 9-7
Required Configurations for Integrating EBS 9-8
How to Prepare OID for Use Without Single Sign-On 9-8
How to Create a User in EBS and Assign a Responsibility 9-9
How to Configure the EBS Applications Profile Options 9-11
How to Add the WebCenter Host as a Trusted Portal Using AutoConfig 9-12
How to Integrate EBS Applications as WSRP Portlets 9-13
How to Prepare the EBS Portlet for Remote Access 9-13
How to Integrate EBS Applications 9-15
How to Integrate EBS Applications as Data Controls 9-16
How to Generate the WSDL 9-17
How to Add a Web Service Data Control to a Portal Page 9-18
Integrating JD Edwards Applications 9-19
How to Prepare the JD Edwards Application for Remote Access 9-19

How to Register the Producer 9-20
How to Add the JD Edwards Portlet to a WebCenter Portal Page 9-20
How to Test the Portlet Connection 9-21
Integrating PeopleSoft Applications 9-21
About Integrating PeopleSoft Applications 9-21
Understanding PeopleSoft Integration 9-21
Requirements for Integrating PeopleSoft Applications 9-21
How to Integrate PeopleSoft Applications as WSRP Portlets 9-22
How to Prepare the PeopleSoft Application for Remote Access 9-22
How to Configure WS-Security for PeopleTools 8.52 and Later 9-25
How to Attach a WS-Security Policy to WebCenter Portal 9-30
How to Integrate PeopleSoft Applications in WebCenter Portal 9-34
How to Configure WS-Security for PeopleTools 8.51 9-35
How to Integrate PeopleSoft Applications as Data Controls in WebCenter Portal 9-38
How to Prepare the WSDL 9-38
How to Create a Web Service Data Control 9-43
Integrating Oracle Business Intelligence Presentation Services 9-45
About Integrating Oracle Business Intelligence Presentation Services 9-45
Understanding Oracle Business Intelligence Presentation Services Integration 9-45
Requirements for Integrating Oracle Business Intelligence Presentation Services 9-45
Advanced Integration Options 9-46
How to Configure Credentials for Connecting to the Oracle BI Presentation Catalog 9-46
How to Check for the BIImpersonateUser 9-47
How to Create the BIImpersonateUser 9-48
How to Grant Permissions to BIImpersonateUser 9-49
How to Integrate Oracle Business Intelligence Objects in WebCenter Portal 9-50
How to Add or Modify a Presentation Services Connection After Deployment 9-51
How to Add Oracle BI Objects to a WebCenter Portal Resource Catalog 9-52
How to Add Oracle BI Content at Runtime 9-53
How to Modify a Business Intelligence Object's Prompt Values 9-54
How to Modify a Business Intelligence Task Flow's Initialization Parameters 9-55
Integrating with Oracle Content and Experience Cloud 9-55
About Oracle Content and Experience Cloud Integration 9-55
Integrating Oracle Content and Experience Cloud with WebCenter Portal 9-56
Creating a Default Oracle Content and Experience Cloud Connection Using WLST 9-56

10 Configuring Search in WebCenter Portal

Integrating with Elasticsearch 10-2
Understanding Search with Elasticsearch 10-2
Prerequisites for Configuring Elasticsearch 10-2

Configuration Roadmap for Elasticsearch in WebCenter Portal 10-3
Installing Elasticsearch and Plugins 10-3
Setting Up Single Server 10-3
Setting Up Cluster and Load Balancing Using OHS 10-7
Securing Elasticsearch Using HTTPS 10-11
Integrating with OCI Search Service with OpenSearch 10-13
Understanding OCI Search Service with OpenSearch 10-13
Prerequisites for Configuring OCI Search Service with OpenSearch 10-14
Configuration Roadmap for OCI Search Service with OpenSearch in WebCenter Portal 10-15
Create a OpenSearch Instance in OCI 10-15
Import OpenSearch SSL Certificate into WebCenter Portal Keystore 10-19
Creating a Crawl Admin User in WebCenter Portal 10-20
Configuring WebCenter Content for Search 10-22
Creating a Crawl User in WebCenter Content 10-22
Configuring the SESCrawlerExport Component 10-24
Configuring WebCenter Portal for Search 10-26
Configuring WebCenter Portal for Elasticsearch 10-26
Configuring WebCenter Portal for OCI Search Service with OpenSearch 10-28
Synchronizing Users in WebCenter Portal 10-29
Configuring Search Crawlers 10-29
Creating a Portal Crawl Source 10-30
Creating a Documents Crawl Source 10-32
Taking a Snapshot of the Content 10-34
Creating a Discussions Crawl Source 10-35
Enabling AutoSuggest in WebCenter Portal 10-37
Configuring Search Custom Attributes 10-37
Creating Custom Facets 10-41
Adding a Custom Metadata Field in Oracle WebCenter Content 10-43
Rebuilding the WebCenter Content Index 10-43
Configuring the SESCrawlerExport Component 10-44
Configuring the Search Setting Metadata 10-45
Configuring the Search Custom Facet Metadata 10-46
Update the Metadata for the Document in Your Portal 10-47
Configuring the File Exclusion in Search Results 10-50
Customizing Search Query Pattern 10-51
Scheduling a Crawl 10-54
Scheduling an Incremental Crawl 10-55
Enabling and Disabling a Scheduled Crawl 10-56
Manually Starting a Full Crawl 10-56
Manually Starting a Fast Crawl 10-57
Manually Starting a Reindex Crawl 10-58

Manually Starting an Incremental Crawl 10-59
Customizing Search Settings in WebCenter Portal Administration 10-59
Modifying the Default Connection Settings for Document Content Crawl Plugin in Elasticsearch Server 10-62
Modifying Search Global Attributes 10-63

11 Managing Instant Messaging and Presence

About Instant Messaging and Presence Connections 11-1
Instant Messaging and Presence Server Prerequisites 11-2
Microsoft Lync - Installation 11-2
Microsoft Lync - Configuration 11-2
Simple Deployment 11-2
Remote Deployment 11-4
Microsoft Lync - Security Considerations 11-10
Registering Instant Messaging and Presence Servers 11-10
Registering Instant Messaging and Presence Servers Using Fusion Middleware Control 11-10
Registering Instant Messaging and Presence Servers Using WLST 11-13
Choosing the Active Connection for Instant Messaging and Presence 11-13
Choosing the Active Connection for Instant Messaging and Presence Using Fusion Middleware Control 11-13
Choosing the Active Connection for Instant Messaging and Presence Using WLST 11-14
Modifying Instant Messaging and Presence Connection Details 11-14
Modifying Instant Messaging and Presence Connections Details Using Fusion Middleware Control 11-14
Modifying Instant Messaging and Presence Connections Details Using WLST 11-15
Deleting Instant Messaging and Presence Connections 11-15
Deleting Instant Messaging and Presence Connections Using Fusion Middleware Control 11-15
Deleting Instant Messaging and Presence Connections Using WLST 11-16
Setting Up Instant Messaging and Presence Defaults 11-16
Testing Instant Messaging and Presence Connections 11-17

12 Managing Mail

About Mail Server Connections 12-2
Configuration Roadmap for Mail 12-2
Mail Server Prerequisites 12-3
Mail Server - Installation 12-3
Mail Server - Configuration 12-3
Configuring Microsoft Exchange Server 2007, 2010, or 2013 for WebCenter Portal 12-4
Mail Server - Security Considerations 12-5

Mail Server - Limitations 12-6
Registering Mail Servers 12-6
Registering Mail Servers Using Fusion Middleware Control 12-6
Registering Mail Servers Using WLST 12-11
Choosing the Active (or Default) Mail Server Connection 12-12
Choosing the Active (or Default) Mail Server Connection Using Fusion Middleware Control 12-12
Choosing the Active (or Default) Mail Server Connection Using WLST 12-13
Modifying Mail Server Connection Details 12-13
Modifying Mail Server Connection Details Using Fusion Middleware Control 12-13
Modifying Mail Server Connection Details Using WLST 12-14
Deleting Mail Server Connections 12-15
Deleting a Mail Connection Using Fusion Middleware Control 12-15
Deleting a Mail Connection Using WLST 12-15
Setting Up Mail Defaults 12-16
Testing Mail Server Connections 12-16
Configuring Send Mail Notifications for WebCenter Portal 12-16
Enabling Shared Mail Connections for Send Mail Notifications 12-18

13 Managing People Connections

About the People Connections Service 13-1
People Connections Prerequisites 13-2
Configuring People Connections for WebCenter Portal 13-2
Accessing People Connections Administrative Settings 13-2
Configuring Activity Stream 13-3
Configuring Connections 13-6
Configuring Profile 13-7
Configuring Message Board 13-10
Configuring Feedback 13-12
Archiving the Activity Stream Schema 13-14
Specifying a Management Chain for Organization View 13-14
Example Embedded LDAP Configuration 13-15
Setting Profile Configuration Properties 13-20
Synchronizing Profiles with the Identity Store 13-21

14 Managing RSS

About RSS 14-1
RSS Prerequisites 14-1
Setting Up a Proxy Server 14-2
Setting Up a Proxy Server for External RSS News Feeds 14-2

Setting Up a Proxy Server Using Fusion Middleware Control 14-2
Setting Up a Proxy Server Using WLST 14-2
Testing External RSS News Feed Connections 14-3

15 Managing Subscriptions and Notifications

About Subscriptions and Notifications 15-1
Setting Up Default Subscription Preferences 15-2
About Subscription Defaults 15-2
Setting Subscription Defaults 15-4
Setting Subscriptions Preferences in WebCenter Portal 15-7
Setting Up Notifications 15-8
About Connection Channels 15-8
Notification Prerequisites 15-9
Installation 15-9
Configuration 15-9
Security 15-10
Limitations 15-10
Configuration Roadmap for Notifications 15-10
Specifying the Notifications Channel Using Fusion Middleware Control 15-12
Specifying the Notifications Channel Using WLST 15-13
Example - Setting Up Mail Notifications for WebCenter Portal Using WLST 15-13
Creating and Applying Custom Notification Templates 15-14
About Overwriting Default Notification Templates 15-14
Overwriting a Default Notifications Template 15-17
Testing the Notifications Connection 15-18

16 Managing the SOA Connection for WebCenter Portal Membership Workflows

Configuration Roadmap for WebCenter Portal Workflows 16-2
About BPEL Connections 16-3
BPEL Server Prerequisites 16-3
BPEL Server - Installation and Configuration 16-4
BPEL Server - Security Considerations 16-4
Specifying the BPEL Server Hosting WebCenter Portal Workflows 16-5
Configuring WebCenter Portal Workflow Notifications to be Sent by Email 16-7
Excluding Webcenter Portal Workflows URL in OAM 16-8

17 Managing Portlet Producers

About Portlet Producers 17-1
Managing WSRP Portlet Producers 17-3
Registering WSRP Producers 17-3
Registering a WSRP Producer Using Fusion Middleware Control 17-4
Registering a WSRP Producer Using WLST 17-5
Registering a WSRP Portlet Producer in WebCenter Portal 17-5
Adding a Grant to the Policy Store for a Mapped User Identity 17-7
WSRP Producer Connection Parameters 17-7
WSRP Producer Security Connection Parameters 17-9
WSRP Producer Keystore Connection Parameters 17-12
Testing WSRP Producer Connections 17-13
Editing WSRP Producer Registration Details 17-13
About Editing WSRP Producer Registration Details 17-14
Editing WSRP Producer Registration Details Using Fusion Middleware Control 17-14
Editing Producer Registration Details Using WLST 17-14
Editing WSRP Producer Registration Details in WebCenter Portal 17-15
Migrating WSRP Producer Metadata to a New WSDL URL 17-16
Editing the Portlet Client Configuration 17-16
Deregistering WSRP Portlet Producers 17-17
About Deregistering Portlet Producers 17-17
Deregistering a WSRP Portlet Producer Using Fusion Middleware Control 17-18
Deregistering a WSRP Portlet Producer Using WLST 17-18
Deregistering a WSRP Portlet Producer in WebCenter Portal 17-19
Managing Oracle PDK-Java Portlet Producers 17-19
Registering an Oracle PDK-Java Portlet Producer 17-19
Registering an Oracle PDK-Java Portlet Producer Using Fusion Middleware Control 17-19
Registering an Oracle PDK-Java Portlet Producer Using WLST 17-20
Registering an Oracle PDK-Java Portlet Producer in WebCenter Portal 17-20
Testing Oracle PDK-Java Producer Connections 17-21
Editing Oracle PDK-Java Portlet Producer Registration Details 17-21
Editing Oracle PDK-Java Portlet Producer Registration Details Using Fusion Middleware Control 17-21
Editing Oracle PDK-Java Portlet Producer Registration Details Using WLST 17-22
Editing Oracle PDK-Java Portlet Producer Registration Details in WebCenter Portal 17-22
Deregistering an Oracle PDK-Java Portlet Producer 17-23
About Deregistering an Oracle PDK-Java Portlet Producer 17-23
Deregistering an Oracle PDK-Java Portlet Producer Using Fusion Middleware Control 17-24
Deregistering an Oracle PDK-Java Portlet Producer Using WLST 17-24

Deregistering an Oracle PDK-Java Portlet Producer in WebCenter Portal 17-24
Oracle PDK-Java Portlet Producer Connection Parameters 17-24
Deploying Portlet Producer Applications 17-27
Preparing Portlet Producer Applications for Deployment 17-27
Deploying a Portlet Producer Application Using Fusion Middleware Control 17-28
Deploying a Portlet Producer Application Using Oracle WebLogic Server Administration Console 17-30
Deploying a Portlet Producer Application Using WLST 17-31
Deploying a Portlet Producer Application Using Oracle JDeveloper 17-33

18 Managing External Applications

About External Applications 18-1
Registering External Applications 18-2
Registering External Applications Using Fusion Middleware Control 18-3
Registering External Applications Using WLST 18-8
Modifying External Application Connection Details 18-8
Modifying External Application Connection Using Fusion Middleware Control 18-8
Modifying External Application Connection Using WLST 18-8
Deleting External Application Connections 18-9
Deleting External Application Connections Using Fusion Middleware Control 18-9
Deleting External Application Connections Using WLST 18-9
Managing External Applications at Runtime 18-10
Registering External Applications at Runtime 18-10
Editing and Deleting External Applications at Runtime 18-11

19 Managing REST Services

About REST Services 19-1
Performing Required Manual Configurations to Enable REST 19-1
Configuring an Identity Asserter 19-2
Configuring the WebLogic Server Credential Store 19-2
Understanding Security Tokens 19-2
Changing the REST Root Name 19-3
Using Compression 19-3
Handling Authentication 19-3

Part IV Monitoring

20 Monitoring WebCenter Portal Performance

Understanding Oracle WebCenter Portal Performance Metrics 20-1
- Understanding Oracle WebCenter Portal Metric Collection 20-2
  - Metric Collection: Since Startup 20-2
  - Metric Collection: Recent History 20-3
  - Metric Collection: Last 'N' Samples 20-4
- Understanding the Key Performance Metrics 20-4
- Using Key Performance Metric Data to Analyze and Diagnose System Health 20-5
- Understanding Some Common Performance Issues and Actions 20-11
- Understanding Page Request Metrics 20-12
  - Understanding Full Page and Partial Page Metrics 20-12
  - Recent Page Metrics 20-13
  - Overall Page Metrics 20-15
- Understanding Portlet Producer Metrics 20-18
  - Recent Portlet Metrics 20-18
  - Overall Portlet Producer Metrics 20-21
  - Overall Portlet Metrics 20-24
- Understanding WebLogic Server Metrics 20-28
  - WebLogic Server Metrics Section 20-30
  - Recent CPU and Memory Usage Section 20-32
  - Recent Session and Thread Usage Section 20-33
  - Recent JDBC Usage Section 20-33
  - Health Metrics Section 20-33
- Understanding Security Metrics 20-35
- Understanding Page Response and Load Metrics 20-35
- Understanding Portal Metrics 20-36
- Understanding Tool and Service Metrics 20-39
  - Metrics Common to all Tools and Services 20-39
  - Metrics Specific to a Particular Tool or Service 20-44
  - Troubleshooting Common Issues with Tools and Services 20-72

Viewing Performance Metrics Using Fusion Middleware Control 20-76
- Monitoring Recent Performance Metrics for WebCenter Portal 20-77
- Monitoring Portal Metrics 20-78
- Monitoring Page Metrics for WebCenter Portal 20-78
- Monitoring Service Metrics for WebCenter Portal 20-79
- Monitoring All Metrics Through the Metrics Palette 20-79

Customizing Key Performance Metric Thresholds and Collection 20-80
- Understanding Customization Options for Key Performance Metrics 20-80
- Understanding Default Metric Collection and Threshold Settings 20-81
- Configuring Thresholds for Key Metrics 20-82

Configuring the Frequency of WebLogic Server Health Checks 20-84
Configuring the Number of Samples Used to Calculate Key Performance Metrics 20-84
Editing Thresholds and Collection Options for WebCenter Portal 20-85
Diagnosing and Resolving Performance Issues with Oracle WebCenter Portal 20-86
Tuning Oracle WebCenter Portal Performance 20-87
Monitoring Performance Using WebCenter Portal Performance Pack 20-87
Improving Data Caching Performance 20-87
Summary of Coherence Cache Types 20-87
Default Coherence Caches in WebCenter Portal 20-88
Overriding the Default Configuration 20-89

21 Managing WebCenter Portal Logs

Introduction to Diagnostic Logging 21-1
WebCenter Portal Diagnostics Log 21-2
Oracle WebCenter Portal Message IDs 21-2
Out-Of-Bound Conditions for Oracle WebCenter Portal Performance Metrics 21-4
Viewing and Configuring Log Information 21-4
Viewing and Configuring WebCenter Portal Logs 21-4
Viewing and Configuring Error Messages in WebCenter Portal 21-5

22 Managing WebCenter Portal Audit Logs

Introduction to Managing Audit Logs 22-1
Configuring Audit Logging 22-2
Setting the Logging Level 22-2
Configuring the Audit Store Database 22-3
Viewing WebCenter Portal Audit Events 22-3
Using WebCenter Portal Audit Logs 22-3
Querying the Audit Schema 22-4

Part V Administering Security

23 Managing WebCenter Portal Security

Introduction to Application Security 23-1
Default Security Configuration 23-4
Administrator Accounts 23-4
Application Roles and Enterprise Roles 23-5
Default Identity and Policy Stores 23-5
Default Policy Store Permissions and Grants 23-6

Permission-based Authorization 23-6
Role-mapping Based Authorization 23-6
Default Policy Store Permissions for WebCenter Portal 23-7
Default Code-based Grants 23-7
Post-deployment Security Configuration Tasks 23-7
Setting the Policy Store Refresh Interval and Other Cache Settings 23-9
Setting the Policy Store Refresh Interval 23-9
Setting the Connection Pool Cache 23-9
Setting User Cache Settings 23-10
Setting Group Cache Settings 23-10

24 Configuring the Identity Store

Reassociating the Identity Store with an External LDAP Server 24-2
Configuring the GUID Attribute for External LDAP Identity Stores 24-5
Adding Users to the Embedded LDAP Identity Store 24-7
Adding Users to the Identity Store Using the WLS Administration Console 24-8
Adding Users to the Identity Store Using an LDIF File 24-9
Enable External LDAP Access 24-9
Create an LDIF File 24-9
Add the Users 24-11
Moving the Administrator Account to an External LDAP Server 24-12
Migrating the Discussions Server to Use an External LDAP 24-13
Changing the Administrator Group Name 24-17
Configuring Oracle WebCenter Content to Share the WebCenter Portal Identity Store LDAP Server 24-18
Aggregating Multiple Identity Store LDAP Servers Using libOVD 24-18
Configuring libOVD for Identity Stores with Complete User Profiles 24-19
Configuring libOVD for Identity Stores with Partial User Profiles 24-20
Restoring the Single Authenticator 24-21
Configuring Dynamic Groups for WebCenter Portal 24-22
Creating a Dynamic Group Using an LDIF File 24-22
Creating a Dynamic Group Using the Oracle Directory Services Manager 24-23
Configuring the REST Service Identity Asserter 24-24
Understanding the REST Service Instance and Identity Asserter 24-24
Setting up the Client Application 24-25
Configuring the WLS Trust Service Asserter 24-27

25 Managing Users and Application Roles

Granting the WebCenter Portal Administrator Role 25-1
Granting the WebCenter Portal Administrator Role Using Fusion Middleware Control 25-2

Granting the WebCenter Portal Administrator Role Using WLST 25-3
Granting Application Roles 25-4
Granting Application Roles Using Fusion Middleware Control 25-4
Granting Application Roles Using WLST 25-6
Using the Runtime Administration Pages 25-7
Configuring Self-Registration By Invitation in WebCenter Portal 25-7

26 Configuring Single Sign-On

Introduction to Single Sign-On 26-1
Configuring Oracle Access Manager 26-2
OAM Components and Topology 26-2
Roadmap to Configuring OAM 26-5
Installing and Configuring OAM 11g 26-6
Installing and Configuring OAM 11g 26-6
Installing and Configuring Oracle HTTP Server 26-6
Configuring Oracle HTTP Server WebGate 26-9
Registering the WebGate Agent 26-11
Configuring the WebLogic Domain for OAM 26-15
Configuring the Oracle Internet Directory Authenticator 26-15
Configuring the OAM Identity Asserter 26-16
Configuring the Default Authenticator and Provider Order 26-17
Adding an OAM Single Sign-on Provider 26-17
Additional Single Sign-on Configurations 26-18
Configuring WebCenter Portal for SSO 26-18
Configuring the Discussions Server for SSO 26-18
Configuring SOA Server Connections for SSO 26-19
Configuring OAM for RSS Feeds Using External Readers 26-20
Configuring the WebLogic Server Administration Console and Enterprise Manager for OAM 11g 26-20
Configuring Elasticsearch for SSO 26-22
Configuring Content Server for SSO 26-23
Restricting Access with Connection Filters 26-23
Configuring Portlet Producers and Additional Components 26-24
Testing Your OAM Installation 26-24
Configuring SAML-based Single Sign-On 26-25
SAML Components and Topology 26-25
Configuring SAML1.1-based Single Sign-On 26-28
SAML Single Sign-on Prerequisites 26-28
Configuring SAML-based SSO 26-31
Configuring SAML SSO for RSS Using External Readers 26-39
Checking Your Configuration 26-39

Disabling Your SAML SSO Configuration 26-40
Removing Your SAML SSO Configuration 26-40
Configuring SAML 2.0-based Single Sign-On 26-41
Creating SAML 2.0 Credential Mapping Provider 26-43
Configuring SAML 2.0 Identity Provider Services 26-45
Configure SAML 2.0 General Services for Identity Provider 26-46
Configuring Service Provider Partner Metadata on SAML Identity Provider Source Site 26-49
Creating SAML 2.0 Identity Assertion Provider 26-50
Configuring SAML 2.0 Service Provider Services 26-51
Configuring SAML 2.0 General Services for Service Provider 26-52
Configuring Identity Provider Metadata on SAML Service Provider 26-54
Troubleshooting Common Issues with SAML 2.0 26-56
Configuring SSO for Microsoft Clients 26-57
Microsoft Client SSO Concepts 26-57
System Requirements 26-59
Configuring Microsoft Clients 26-60
Configuring the Negotiate Identity Assertion Provider 26-61
Configuring an Active Directory Authentication Provider 26-62
Configuring WebCenter Portal 26-64
Configuring the Discussions Server for SSO 26-64
Configuring SSO with Virtual Hosts 26-64
Understanding the Need for a Virtual Host 26-65
Configuring Virtual Hosts for OAM 11g 26-65

27 Configuring SSL

Securing the Browser Connection to WebCenter Portal using SSL 27-2
Creating the Custom Keystore 27-2
Configuring the Custom Identity and Custom Trust Keystores 27-4
Configuring the SSL Connection 27-5
Securing the Connection from Oracle HTTP Server to WebCenter Portal with SSL 27-5
Wiring the WebCenter Portal Ports to the HTTP Server 27-6
Configuring the SSL Certificates 27-7
Securing the Browser Connection to Discussions with SSL 27-8
Creating the Custom Keystore for Discussions 27-8
Configuring the Identity and Trust Keystore for Discussions 27-10
Configuring and Securing the SSL Connection for Discussions 27-11
Securing the WebCenter Portal Connection to Portlet Producers with SSL 27-11
Creating the Custom Keystores for Portlet Producers 27-12
Configuring the Identity and Trust Keystores for Portlet Producers 27-13
Configuring the SSL Connection for Portlet Producers 27-14

Registering the SSL-enabled WSRP Producer and Running the Portlets 27-14
Securing the WebCenter Portal Connection to the LDAP Identity Store 27-15
Exporting the OID Certificate Authority (CA) 27-16
Enabling the SSL in OID 27-16
Importing the OID Certificate 27-17
Establishing the SSL Connections 27-18
Securing the WebCenter Portal Connection to IMAP and SMTP with SSL 27-20
Securing the WebCenter Portal Connection to an External BPEL Server with SSL 27-21
Securing the WebCenter Portal Connection to Content Server with SSL 27-22
Configuring Content Server SSL for Socket SSL 27-22
Creating Content Server SSL Connection in WebCenter Portal 27-22

28 Configuring Web Services Security

Configuring WS-Security for a Typical Topology 28-2
Creating the WebCenter Portal Domain Keystore 28-2
Creating the SOA Domain Keystore 28-4
Configuring the Discussions Server 28-8
Attaching Security Policies for WebCenter Portal and Discussions Web Service Endpoints 28-8
Securing the Discussions End Points 28-9
Configuring the Discussions Server Connection Settings 28-15
Configuring WS-Security for Multiple Domains 28-15
Setting Up the WebCenter Portal Domain Keystore 28-16
Creating the SOA Domain Keystore 28-16
Configuring an External Discussions Server 28-16
Securing the Discussions Service End Points 28-16
Creating the Discussions Server Keystore 28-16
Configuring the Discussions Server Connection Settings 28-19
Creating the External Portlet Domain Keystore 28-20
Securing WebCenter Portal for Applications Consuming WebCenter Portal Client API with WS-Security 28-21
Configuring a Typical Topology for Applications Consuming WebCenter Portal Client API 28-22
Configuring a Multiple Domain Topology for Applications Consuming the WebCenter Portal Client API 28-22
JKS Command Summary for a Typical Topology 28-23
JKS Command Summary for Extensions to a Typical Topology 28-24

29 Configuring Security for Portlet Producers

Securing a WSRP Producer 29-1

Deploying the Producer 29-1
Attaching a Policy to the Producer Endpoint 29-1
Setting Up the Keystores 29-5
Securing a PDK-Java Producer 29-5
Defining a Shared Key as a Password Credential 29-5
Defining a Shared Key Using Fusion Middleware ControlFusion Middleware Control 29-6
Defining a Shared Key Using WLST 29-6
Registering an Oracle PDK-Java Producer with a Shared Key 29-7

30 Managing Impersonation

Introduction to WebCenter Portal Impersonation 30-1
About WebCenter Portal Impersonation 30-1
Best Practices for Using WebCenter Portal Impersonation 30-2
Preparing WebCenter Portal for Impersonation 30-2
WebCenter Portal Impersonation Requirements 30-3
Turning on Impersonation in OAM 30-3
Adding Impersonation Attributes to the Identity Store 30-3
Adding Impersonation Attributes for Individual Users 30-4
Adding Impersonation Attributes for Multiple Users 30-4
Configuring WebCenter Portal for Impersonation 30-5
Configuring Impersonators 30-6
Disabling Impersonation 30-7
Turning off the Session Indicator 30-8
Overriding the Impersonation Hotkey 30-9
Managing Audit Logs for WebCenter Portal Impersonation 30-9

Part VI Administering WebCenter Portal Lifecycle

31 Understanding the WebCenter Portal Lifecycle

What Is the WebCenter Portal Life Cycle? 31-1
What Are the Major WebCenter Portal Lifecycle Tasks? 31-4
One-Time Setup Tasks 31-4
Understanding WebCenter Portal Staging and Production Environments 31-4
Lifecycle Tasks 31-6
Permissions Required to Perform WebCenter Portal Lifecycle Operations 31-8
Managing Security Through the WebCenter Portal Lifecycle 31-10

32 Deploying Portals, Templates, Assets, and Extensions

Deploying Portals 32-1
About Portal Deployment 32-2
Directly Deploying Portals Using WebCenter Portal 32-5
Creating a Portal Server Connection 32-5
Deploying a Portal Using WebCenter Portal 32-6
Viewing Portal Deployment History 32-8
Directly Deploying Portals Using WLST 32-9
Step 1: Complete Prerequisites for Direct Portal Deployment 32-9
Step 2: Run deployWebCenterPortal in the Source Environment 32-10
Step 3: Verify Newly Deployed Portal in the Target Environment 32-10
Deploying Portal Archives 32-11
Understanding Portal Archives 32-12
Securing Archives 32-18
Exporting and Importing Portal Archives 32-20
Exporting Portals to an Archive 32-20
Importing Portals from an Archive 32-25
Viewing and Extracting Portal Archives 32-32
Deploying Portal Templates 32-32
Exporting Portal Templates 32-33
Exporting Portal Templates to an Archive Using WebCenter Portal 32-33
Exporting Portal Templates to an Archive Using WLST 32-33
Importing Portal Templates 32-34
Importing Portal Templates from an Archive Using WebCenter Portal 32-34
Importing Portal Templates from an Archive Using WLST 32-34
Deploying Assets 32-35
Exporting Assets, Devices, and Device Groups to an Archive 32-36
Exporting Assets to an Archive from WebCenter Portal 32-36
Exporting Devices and Device Groups to an Archive 32-36
Exporting an Asset, Device, or Device Group to an Archive Using WLST 32-37
Exporting Assets Using REST API 32-38
Importing Assets from an Archive 32-39
About Permissions Required to Import (or Export) Assets 32-39
Importing Assets from an Archive using WebCenter Portal 32-40
Importing Devices and Device Groups Using WebCenter Portal 32-40
Importing Assets from an Archive using WLST 32-40
Importing Assets Using REST API 32-41
Deploying Custom Shared Library Extensions 32-42
Moving Connections Details from Staging to Production 32-42
Exporting WebCenter Portal Connections Details to a File 32-42

Importing New WebCenter Portal Connections from a File 32-43
Migrating Discussions Resources for a Portal 32-44
Exporting Portal Discussions to an Archive 32-44
Importing Portal Discussions from an Archive 32-46
Propagating and Redeploying Portals in Production 32-48
Understanding Portal Propagation 32-49
Propagating Portal Changes Using WebCenter Portal 32-49
Propagating Portal Changes Using WLST 32-52
Redeploying a Portal Using WebCenter Portal 32-52

33 Managing WebCenter Portal Backup, Recovery, and Cloning

Understanding WebCenter Portal Back Up and Recovery 33-1
Comparing Back up, Recovery, and Migration Tools for WebCenter Portal 33-2
Backing Up Individual Portals 33-5
Backing Up Portals Using WLST 33-5
Backing Up Discussions and External Data for a Portal 33-6
Restoring Portals from a Backup 33-6
Restoring Portals from an Archive Using WLST 33-7
Restoring Discussions and External Data for a Portal 33-7
Backing Up an Entire WebCenter Portal Installation 33-8
Backing Up and Restoring All WebCenter Portal Schema Data 33-9
Prerequisites 33-9
Back Up (Export) WebCenter Portal Schema Data 33-9
Restore (Import) WebCenter Portal Data 33-10
Backing Up and Restoring All MDS Schema Data 33-11
Prerequisites 33-11
Back Up (Export) All MDS Schema Data 33-12
Restore (Import) MDS Schema Data 33-13
Backing Up and Restoring All WebCenter Content Data 33-13
Backing up and Restoring Discussion Schema Data 33-14
Prerequisites 33-15
Back Up (Export) All Discussions Schema Data 33-15
Restore (Import) Discussions Schema Data 33-16
Backing up and Restoring Other Schema Data (ACTIVITIES and PORTLET) 33-17
Backing Up and Restoring LDAP Identity Store 33-20
Backing Up and Restoring Policy Stores (LDAP and Database) 33-20
Backing Up and Restoring Credential Stores (LDAP and Database) 33-21
Backing Up and Restoring a WebCenter Portal Domain 33-21
Backing Up and Restoring Portlet Producer Metadata 33-21
Backing Up (Exporting) Portlet Client Metadata 33-22

Restoring (Importing) Portlet Client Metadata 33-22
Backing Up and Restoring Analytics Metadata 33-22
Backing Up and Restoring Audit Repository Configuration 33-22
Migrating Entire WebCenter Portal to Another Target 33-22
Understanding Import and Export for WebCenter Portal 33-23
Prerequisites for WebCenter Portal Export and Import 33-27
Exporting WebCenter Portal to an Archive 33-28
Exporting WebCenter Portal Using Fusion Middleware Control 33-29
Exporting WebCenter Portal Using WLST 33-30
Importing a WebCenter Portal Archive 33-31
Importing WebCenter Portal Using Fusion Middleware Control 33-31
Importing WebCenter Portal Using WLST 33-32
Verifying WebCenter Portal After Import 33-32
Restoring an Entire WebCenter Portal Installation 33-33
Using Scripts to Back Up and Restore WebCenter Portal 33-34
Understanding Back Up and Restore Script Files 33-34
master_script.sh 33-35
wlst_script.py 33-40
backup.properties and restore.properties Files 33-42
Using Scripts to Back Up WebCenter Portal 33-48
Create Back Up Scripts 33-49
Complete Prerequisite Tasks for Security Store Back Up 33-49
Set Back Up Parameters and Customize Scripts 33-50
Run the Back Up Script 33-51
Verify Back Up Archives 33-51
Schedule Regular Back Ups Using the Scripts 33-52
Restoring WebCenter Portal from Backups Using Scripts 33-52
Create Restore Scripts 33-52
Restore Database Schemas Manually 33-52
Complete Prerequisite Tasks for Security Store Restore 33-54
Set Restore Script Parameters 33-55
Run the Restoration Script 33-55
Verify Restored Data 33-55
Cloning a WebCenter Portal Environment 33-56

Part VII Administering Multilanguage Portals

34 Managing a Multilanguage Portal

About Languages in WebCenter Portal 34-1
Languages Supported Out-of-the-Box by WebCenter Portal 34-3

Translating Strings for a Portal 34-4
Translating Strings for Page Content 34-6
Adding Support for a New Language to WebCenter Portal 34-9
Translating Strings for Search Facets 34-9
Translating Strings for Custom Search Attributes 34-12

Part VIII Administering Portals in WebCenter Portal

35 Exploring the Settings Pages in WebCenter Portal Administration

Working with WebCenter Portal Administration Settings 35-1
Accessing the Settings Pages in WebCenter Portal Administration 35-3

36 Exploring the Portals Page in WebCenter Portal Administration

About the Portals Page in WebCenter Portal Administration 36-1
Accessing the Portals Page in WebCenter Portal Administration 36-2
Sorting the Portals Listing 36-4
Creating a Portal 36-5
Exporting and Importing a Portal 36-5
Viewing Information About Any Portal 36-6
Sharing the Link to a Portal 36-7
Closing Any Portal 36-8
Reactivating Any Portal 36-9
Taking Any Portal Offline 36-10
Bringing Any Portal Back Online 36-11
Deleting a Portal 36-12

37 Configuring Global Defaults Across Portals

Customizing the Name and Logo in the Home Portal 37-2
Choosing a Default Page Template 37-3
Choosing a Default Skin 37-5
    Applying a Skin for WebCenter Portal 37-6
Choosing Default Resource Catalogs 37-7
Customizing Copyright and Privacy Statements 37-9
Customizing the Online Help Link 37-10
Choosing a Default Display Language 37-12
    Customizing the Language List 37-13
Redirecting on Logout 37-15
Choosing a Default Start (or Landing) Page 37-16

Specifying a Default Start Page for Groups 37-17
Specifying a Default Start Page for Authenticated Users 37-19
Specifying a Default Start Page for Public Users 37-21
Specifying Session Timeout Settings 37-22
Enabling Self-Registration 37-25
About Self-Registration 37-25
Enabling Anyone to Self-Register 37-27
Enabling Self-Registration By Invitation-Only 37-28
Choosing a Default Look and Feel for New Pages 37-29
Enabling and Disabling Access to the Home Portal 37-30
Setting Up Defaults for WebCenter Portal Tools and Services 37-31
Adding a List of Valid External URLs 37-32

38 Managing Security Across Portals

About WebCenter Portal Security 38-1
About Users 38-4
About Application Roles and Permissions 38-5
About Application Roles 38-5
Default Application Roles 38-5
Custom Application Roles 38-8
About Application Permissions 38-9
Understanding Application Permissions 38-9
Default Application Permissions Assignments to Application Roles 38-12
Understanding Discussion Server Role Mapping 38-14
Understanding Enterprise Group Role Mapping 38-15
About Roles and Permissions Within a Portal 38-16
Managing Users 38-16
Adding and Removing Users 38-17
Assigning Users (and Groups) to Application Roles 38-18
Assigning a User to a Different Application Role 38-20
Revoking Application Roles 38-22
Managing Application Roles and Permissions 38-23
Viewing Application Roles and Permissions 38-23
Defining Application Roles 38-25
Modifying Application Role Permissions 38-26
Granting Permissions to the Public-User 38-27
Granting Permissions to the Authenticated-User 38-27
Granting Permissions to the Portal Creator 38-28
Deleting Application Roles 38-28

39 Working with Global Attributes Across Portals

About Global Attributes 39-1
Adding a Global Attribute 39-2
Editing a Global Attribute 39-3
Deleting a Global Attribute 39-4

40 Customizing System Pages

About System Pages 40-2
About Built-In System Pages 40-2
Customizing System Pages for All Portals 40-6
Creating a Page Variant of a System Page for Device Groups 40-7
Managing a Page Variant of a System Page for Device Groups 40-11
Setting System Page Properties 40-12
Removing All Page Customizations from a System Page 40-16

41 Managing Business Role Pages

About Business Role Pages 41-1
About Built-In Business Role Pages 41-2
Setting Page Creation Defaults for Business Role Pages 41-3
Creating a Business Role Page 41-5
Specifying the Target Audience for a Business Role Page 41-7
Setting Access on a Custom Business Role Page 41-8
Providing Public Access to a Custom Business Role Page 41-12
Setting Access on a Built-in Business Role Page 41-13
Revoking Access to a Custom Business Role Page 41-15
Showing and Hiding Business Role Pages 41-15
Setting a Default Display Order for Business Role Pages 41-16
Editing a Business Role Page 41-18
Editing the Source of a Business Role Page 41-19
Copying a Business Role Page 41-20
Removing All User Customizations from a Business Role Page 41-22
Deleting a Custom Business Role Page 41-22

42 Managing Personal Pages

About Personal Page Administration 42-1
Setting Application-Level Page Creation Defaults for Personal Pages 42-2
Preventing Users from Creating Personal Pages 42-2
Providing Navigation to Personal Pages 42-3

Changing Access Permissions on a Personal Page 42-3
Editing a Personal Page 42-6
Editing the Source of a Personal Page 42-7
Copying a Personal Page 42-8
Removing All User Customizations from a Personal Page 42-9
Deleting a Personal Page 42-10

43 Administering Device Settings

About Device Settings 43-1
- Introduction to Device Settings 43-2
- What Are Devices? 43-2
- What Are Device Groups? 43-3
- Other Related Concepts 43-4
- Basic Use Case: Adding Support for a New Device 43-5
- Understanding How Device Settings are Applied 43-6

Creating and Managing Devices 43-7
- Creating a New Device 43-7
- Editing a Device 43-9
- Copying a Device 43-10
- Filtering the List of Devices 43-10
- Deleting a Device 43-10

Creating and Managing Device Groups 43-11
- Creating a Device Group 43-12
- Editing a Device Group 43-13
- Copying a Device Group 43-14
- Showing and Hiding Device Groups 43-15
- Setting a Default Device Group 43-16
- Ordering Device Groups 43-16
- Filtering Device Groups 43-17
- Deleting a Device Group 43-17

Enabling Page Variants for Device Groups 43-18
Managing Device and Device Group Lifecycles 43-19
- Downloading a Device Group or Device 43-19
- Uploading a Device Group or Device 43-20

Previewing Devices 43-20
Guidelines and Best Practices for Device Settings 43-21
Discovering Device Attributes: A Sample Task Flow 43-21

44 Customizing Task Flows

About Task Flow Customization at the Application Level 44-1
Customizing Task Flows at the Application Level 44-2
Removing Task Flow Customizations 44-6

45 Analyzing Portal Usage

About the Analytics Task Flows and Service 45-1
About the Analytics Administration Page 45-2
Working with Analytics Task Flows 45-3
- Understanding Analytics Task Flows 45-3
  - WebCenter Traffic 45-4
  - Page Traffic (Administrator) 45-4
  - Login Metrics (System Administrator) 45-4
  - Portal Traffic (System Administrator) 45-5
  - Portal Response Time (System Administrator) 45-5
  - Portlet Traffic (Administrator) 45-6
  - Portlet Instance Traffic (Administrator) 45-6
  - Portlet Response Time (Administrator) 45-7
  - Portlet Instances Response Time (Administrator) 45-7
  - Search Metrics 45-7
  - Document Metrics (System Administrator) 45-7
  - Wiki Metrics (System Administrator) 45-8
  - Blog Metrics (System Administrator) 45-9
  - Discussion Forum Metrics (System Administrator) 45-9
- Adding Analytics Task Flows to a Page 45-10
- Customizing Analytics Reports 45-10
- Personalizing Your Analytics Report 45-10
  - Report Display Options 45-11
  - Query Options 45-13
- Setting Analytics Task Flow Properties 45-14
  - About the Analytics Service Task Flow Properties 45-15
  - Analytics Service Task Flow Parameters 45-15

Part IX Appendixes

A Managing Announcements and Discussions

- About Discussions Server Connections A-2
- Discussions Server Prerequisites A-2

Discussions Server - Installation A-3
Discussions Server - Configuration A-3
Discussions Server - Security Considerations A-4
Discussions Server - Limitations A-6
Registering Discussions Servers A-7
Registering Discussions Servers Using Fusion Middleware Control A-7
Registering Discussions Servers Using WLST A-10
Choosing the Active Connection for Discussions and Announcements A-11
Choosing the Active Connection for Discussions and Announcements Using Fusion Middleware Control A-11
Choosing the Active Discussion for Discussions and Announcements Using WLST A-11
Modifying Discussions Server Connection Details A-12
Modifying Discussions Server Connection Details Using Fusion Middleware Control A-12
Modifying Discussions Server Connection Details Using WLST A-13
Deleting Discussions Server Connections A-13
Deleting a Discussions Server Connection Using Fusion Middleware Control A-13
Deleting a Discussions Server Connection Using WLST A-13
Setting Up Discussions Defaults A-14
Setting Up Announcements Defaults A-14
Testing Discussions Server Connections A-15
Granting Administrator Permissions on the Discussions Server A-15
Granting Administrator Role on the Discussions Server A-15
Granting the Discussions Server Administrator Role Using WLST A-15
Granting the Discussions Server Administrator Role Using the Admin Console A-15
Revoking the Discussions Server Administrator Role A-16
Configuring Discussion Forum Options for WebCenter Portal A-17
Accessing the Discussions Server Admin Console A-18
Specifying Where Discussions and Announcements are Stored on the Discussions Server A-20
Choosing How Many Discussion Topics to Save In Portal Templates A-22

B Oracle WebCenter Portal Configuration

Configuration Files B-1
adf-config.xml and connections.xml B-1
web.xml B-6
webcenter-config.xml B-7
Cluster Configuration B-8
Configuration Tools B-8
Modifying the File Upload Size in Content Manager B-10

C Third-Party Product Support

D Migrating Wiki Content to WebCenter Portal

Understanding Wiki Documents and Wiki Pages D-1
Understanding Wiki Documents D-1
Understanding Wiki Pages D-2
Migrating Data from the Source Wiki Application to WebCenter Portal D-3
Preparing WebCenter Portal for Importing Wiki Content D-3
Writing and Running a Custom Wiki Extraction Tool to Extract Content from the Wiki Application D-4
Using the Document Migration Utility to Import the Archive into the Target Portal D-12
Creating Wiki Pages in WebCenter Portal for the Content in WebCenter Content Server D-16

E Migrating Folders_g to FrameworkFolders

Understanding Folders_g Migration to FrameworkFolders E-1
Understanding the Folders_g and FrameworkFolders Directory Structure E-1
Migrating WebCenter Portal Data E-3
Migration Roadmap E-3
Running exportFoldersGData to Generate the Pre-Migration Data E-5
Migrating WebCenter Portal MetaData to FrameworkFolders E-6
Running migrateFoldersGDataToFrameworkFolders to Validate the Migrated Data E-9
Troubleshooting Migration Issues E-11

F Troubleshooting WebCenter Portal

Using My Oracle Support for Additional Troubleshooting Information F-1
Troubleshooting Oracle WebCenter Portal Configuration Issues F-2
Configuration Options Unavailable F-2
Logs Indicate Too Many Open Files F-2
Troubleshooting Oracle WebCenter Portal WLST Command Issues F-3
No Oracle WebCenter Portal WLST Commands Work F-3
Connection Name Specified Already Exists F-3
WLST Shell is Not Connected to the WebLogic Server F-4
More Than One Application with the Same Name Exists in the Domain F-4
More Than One Application with the Same Name Exists on a Managed Server F-5
Already in Domain Runtime Tree Message Displays F-5
Troubleshooting Oracle WebCenter Portal Performance Issues F-5
About Performance Monitoring and Troubleshooting Tools F-6

How to Identify Slow Pages F-7
How to Identify Slow Page Components F-7
How to Troubleshoot Slow Page Requests F-11
How to Troubleshooting Requests using JRockit Flight Recordings F-15
Troubleshooting WebCenter Portal Workflows F-17
Email Notifications Not Working F-17
Validating the WebCenter Portal Workflow Configuration F-18
Troubleshooting Issues with WebCenter Portal Workflows F-18
Troubleshooting WebCenter Portal Import and Export F-19
ResourceLimitException Issue F-20
LockRefreshTask Issue F-20
Portals and Portal Templates Not Available After Import F-21
Unable to Migrate Portals or Documents If the Source and Target Applications Share the Same Content Server F-21
Target Portal Server Shown As Unavailable When Creating a Connection F-21
Troubleshooting Individual Portal and Portal Template Import and Export F-21
Portal Blocked After Unsuccessful Export or Import F-22
Page or Portal Not Found Message After Import F-22
Portal Import Archive Exceeds Maximum Upload File Size F-22
Maximum Number of Portals Exceeded on Export F-22
Lists Not Imported Properly F-23
Exporting and Importing Portals with Tools and Services Configured F-23
Tools and Services Disabled After Import F-24
Importing from the Subportals Page F-24
Unable to Import a Portal If the Source and Target Applications Share the Same Content Server F-24
Shared Library Changes Not Available after Portal Deployment F-25
Members Not Listed in an Imported Portal F-25
Deployment Messages Not Displayed in the Browser Locale F-25
Troubleshooting Issues with Mail F-25
Mail is Not Accessible in Secure Mode F-26
Mail is Not Accessible in Non-Secure Mode F-26
Unable to Create Distribution Lists in the Non-Secure Mode F-26
Unable to Create Distribution Lists in the Secure Mode F-27
Provisioning of Mail Fails in a Portal (Default Distribution List not Created) F-27
Unable to Configure the Number of Mail Messages Downloaded F-27
Unable to Publish and Archive WebCenter Portal Mail F-28
Changing Passwords on Microsoft Exchange F-28
Troubleshooting Issues with Announcements and Discussions F-29
Authentication Failed F-29
Discussions Cannot Be Enabled in WebCenter Portal F-29
Login Failed F-30

Login Does Not Function Properly After Configuring Oracle Access Manager F-30
Category Not Found Exceptions F-31
Watched Topics and Recent Topics Not Displaying Topics From Multiple Discussion Forums F-31
Discussion and Announcement Updates Not Displayed F-31
Announcements Page Displays "User Is Not Authorized" F-32
Discussions Page Displays "User Is Not Authorized" F-32
Troubleshooting Issues with Events F-32
Troubleshooting Issues with Users and Roles F-32
Troubleshooting Issues with Content Repositories F-33
Documents Tools Unavailable in WebCenter Portal F-33
Troubleshooting Issues with Analytics F-34
Troubleshooting Issues with Notifications F-35
Troubleshooting External Application Issues F-37
Users Experience Password Lockout F-37
Troubleshooting Security Configuration Issues F-37
WebCenter Portal Application Does Not Find Users in LDAP Provider F-37
Portal Created with Errors When Logged in as OID User F-37
Users Cannot Self-Register when WebCenter Portal Configured with Active Directory F-38
User Made Administrator Does Not Have Administrator Privileges F-38
Deploying the SAML SSO-specific Discussions EAR file Produces an Exception F-39
Configuring SAML Single Sign-on Produces 403 Error F-39
Impersonation Session Produces Error with OAM 11.1.2.2.0 F-40
Troubleshooting Issues with External Links F-40
Troubleshooting Issues with Elasticsearch F-40
Profile Crawling Fails with 401 Error F-41

Preface

This guide explains how to administer Oracle WebCenter Portal, including how to start, stop, and configure WebCenter Portal components, configure back-end servers and security, monitor performance, and also how to back up, recover, and migrate portal deployments and services.

Topics:

- Audience
- Documentation Accessibility
- Related Documents
- Conventions

Audience

This guide is intended for system administrators responsible for configuring Oracle WebCenter Portal. For a complete description of these roles and other WebCenter Portal personas, refer to Who's Who.

This guide assumes that the audience is familiar with the concepts and content described in Administering Oracle Fusion Middleware.

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 that have purchased support 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.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Related Documents

Documentation for Oracle WebCenter Portal is available in the Oracle Fusion Middleware library on the Oracle Help Center.

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.

Who's Who

The WebCenter Portal documentation is organized so that the tasks in a particular guide address a specific user *persona*. Each persona is associated with a set of skills required to work with WebCenter Portal, from basic to advanced.

This preface introduces you to the WebCenter Portal personas and describes the ways in which they might interact with WebCenter Portal. Each persona is assigned a default role provided out-of-the-box with WebCenter Portal. The default roles are 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 that is comprised of the following personas:

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

This guide is aimed at the *system administrator* persona.

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 manager will likely assign Karen a role that includes `View Pages` and `Customize Pages` permissions.

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

Knowledge Worker Tasks in WebCenter Portal

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

- Editing and updating pages for which she has been assigned content contribution permissions
- Connecting to and collaborating with other WebCenter Portal users by sharing information, files, and links; and by interacting through instant messaging, mail, message boards, 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, viewing the activities of the portals she is a member of and users she's connected to, and monitoring WebCenter Portal RSS feeds
- Staying organized through the use of favorites, notes, calendars, lists, links to portal objects, and tags

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

Application Specialist

Ari is an *application specialist* who works in WebCenter Portal 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 `Portal Manager` role to manage the portal.

For more information about roles and permissions, see About Roles and Permissions for a Portal in *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 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 *Building Portals with Oracle WebCenter Portal*. To work with his personal view of the Home portal, Ari will also refer to *Using Portals in 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 WebCenter Portal) 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 page templates

Information targeted to web developers like Wendy is in Creating a Look and Feel for Portals in *Building Portals with Oracle WebCenter Portal*.

Developer

Dave is a *developer* who 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 works with JDeveloper to develop and extend assets for use in WebCenter Portal.

Developer Tasks

Tasks that are typical of a developer like Dave include:

- Developing custom assets such page templates and resource catalogs 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 to developers like Dave is in *Developing for Oracle WebCenter Portal*.

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, 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 About Application Roles and Permissions in *Administering Oracle WebCenter Portal*.

System Administrator Tasks

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

- Uses WebCenter Portal administration to administer all portals (including import and export of portals) and security site-wide
- Uses WebCenter Portal administration to manage site-wide system pages, business role pages, and personal pages
- 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 to system administrators like Syed is in *Administering Oracle WebCenter Portal* and *WebCenter WLST Command Reference Reference*.

Part I
Introduction to Oracle WebCenter Portal

This part of *Administering Oracle WebCenter Portal* provides an introduction to Oracle WebCenter Portal and its administration tools.

- Introduction to Administration for WebCenter Portal

1

Introduction to Administration for WebCenter Portal

With WebCenter Portal, you can create internal and external portals, websites, and composite applications. Begin by exploring the topology, architecture, administrative tools, and tasks involved in setting up WebCenter Portal.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

Topics:

- Introducing Oracle WebCenter Portal
- Oracle WebCenter Portal Architecture
- Oracle WebCenter Portal Topology
- Understanding the Oracle WebCenter Portal Installation
- Understanding Administrative Operations, Roles, and Tools
- Performance Monitoring and Diagnostics
- Understanding Security
- Data Migration, Backup, and Recovery
- Oracle WebCenter Portal Administration Tools

Introducing Oracle WebCenter Portal

Companies use Oracle WebCenter Portal to build enterprise-scale intranet and extranet portals that provide a foundation for the next-generation user experience (UX) with Oracle Fusion Middleware and Oracle Fusion Applications. Portals built with Oracle WebCenter Portal commonly support thousands of users who create, update, and access content and data from multiple back-end sources. Oracle WebCenter Portal delivers intuitive user experiences by leveraging the best UX capabilities from a significant portfolio of leading portal products and related technologies. From the user's perspective, the integration is seamless.

Oracle WebCenter Portal provides users with a personalized, secure, and efficient way of consuming information and interacting with people and applications in the context of business processes. It optimizes the connections between people, information, and applications; provides business activity streams so users can navigate, discover, and access content in

context; and offers dynamic personalization of applications, portals, and sites to provide a customized experience.

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

- Oracle WebCenter Portal Architecture
- Oracle WebCenter Portal Topology

Oracle WebCenter Portal Architecture

Oracle WebCenter Portal comprises the following components:

- WebCenter Portlets
- Application Development Framework
- Portal Composer
- Tools and Services
- Discussion Server
- Analytics

WebCenter Portlets

Develop and integrate portlets into WebCenter Portal:

- Support for JSR-168 and JSR-286 standards-based WSRP portlets
- Oracle JSF Portlet Bridge, which lets you expose JSF pages and Oracle ADF task flows as standards-based portlets

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

Portal Composer

Portal Composer comprises all the browser-based creating, editing, and administration areas of WebCenter Portal:

- A browser-based platform for creating and administering enterprise portals, multiple sites, and communities.

- A Home portal, where users have access to their profile, available portals, portal templates, and documents, and can customize certain elements of their own view of the Home portal.
- A browser-based portal editor, where users can perform runtime portal customization to modify portal settings and create portal pages and device-enabled page variants. An intuitive page editor enables users to modify page layout, properties, wiring, and include components such as task flows, portlets, blogs, wikis, RSS, activity stream, search, and more.

Tools and Services

Table 1-1 lists the tools and services available in WebCenter Portal.

Table 1-1 WebCenter Portal Tools and Services

A Through I	L Through T
Activity Stream	Links
Analytics	Lists
Announcements (Only in upgraded installations)	Mail
Discussions (Only in upgraded installations)	Notes
Documents (includes Wikis and Blogs)	People Connections Polls
Events	RSS
Instant Messaging and Presence (IMP)	Search Tags

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

For more information, see Managing Tools and Services.

Discussion Server

A discussion server is available so you can integrate discussion forums and announcements into your portals. For information, see Managing Announcements and Discussions.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

Analytics

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

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

For information, see Managing Analytics.

Oracle WebCenter Portal Topology

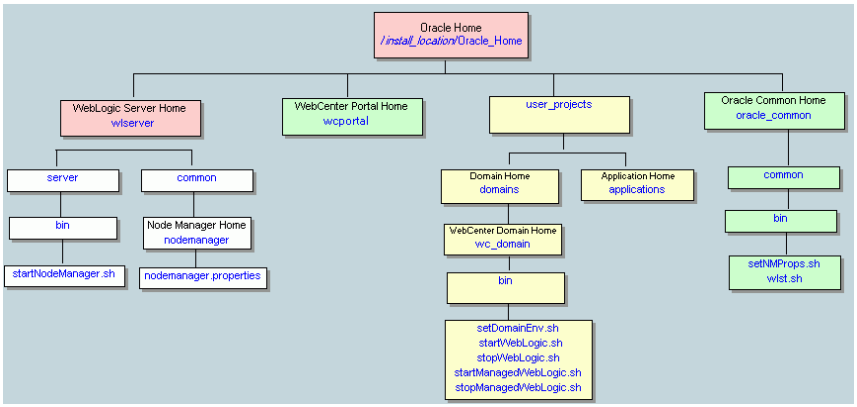
This section describes Oracle WebCenter Portal topology and configuration in the following topics:

- Oracle WebCenter Portal Directory Structure
- Oracle WebCenter Portal Managed Servers
- Oracle WebCenter Portal Configuration Considerations
- Discussions Server Configuration
- Oracle WebCenter Portal State and Configuration Persistence
- Analytics Considerations
- Oracle WebCenter Portal Log File Locations

Oracle WebCenter Portal Directory Structure

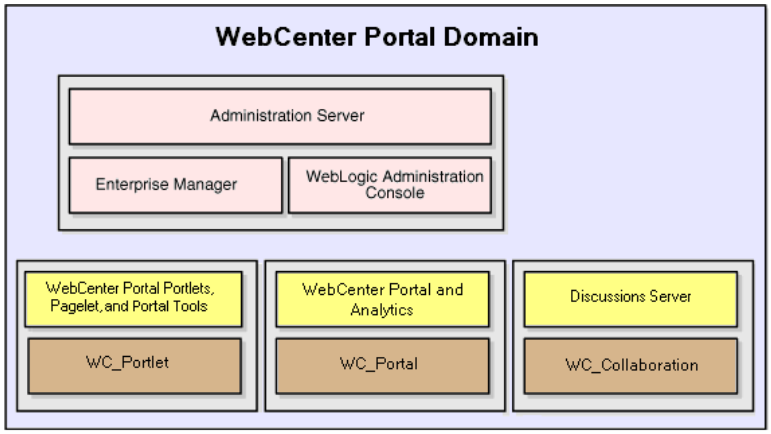
Oracle WebCenter Portal installation creates the WebCenter Portal product home directory (`wcportal`), under the Oracle Home directory, that contains WebCenter Portal binaries and supporting files. The following figure describes directory structure of an Oracle WebCenter Portal installation.

Figure 1-1 Directory Structure of an Oracle WebCenter Portal Installation



The installation also creates a WebCenter Portaldomain (default name `base_domain`), containing the administration server and several managed servers to host various WebCenter Portal components. In Figure 1-2, applications are shown in yellow, while the managed servers they run on are shown in brown.

Figure 1-2 Oracle WebCenter Portal Topology Out-of-the-Box



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

- `WC_Portal`- Hosts WebCenter Portal, Oracle's out-of-the-box portal application, and analytics
- `WC_Portlet` - Hosts out-of-the-box portlets and WebCenter Portal tools
- WC_Collaboration - Hosts the discussions server and any additional services that you choose to integrate

For more information about managed servers, see Understanding Oracle Fusion Middleware Concepts in *Administering Oracle Fusion Middleware*.

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_Portal	WebCenter Portal	webcenter
	WebCenter Portal online help	webcenter-help
	Analytics	analytics-collector
WC_Portlet	WSRP tools	portalTools
		wsrp-tools
WC_Collaboration	Discussions Server	owc_discussions

Oracle WebCenter Portal Configuration Considerations

The main configuration files for WebCenter Portal are listed and described in Table 1-3. Both these files are supplied within the application deployment .EAR file.

Table 1-3 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 is currently using.
`connections.xml`	Stores basic configuration for connections to external services.

WebCenter Portal uses the Oracle Metadata Services (MDS) repository to store its configuration data; it accesses the MDS repository as a JDBC data source within the Oracle WebLogic framework.

The MDS repository stores post deployment configuration changes for WebCenter Portal 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 WebCenter Portal 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 Oracle WebCenter Portal Configuration.

To find out more about the configuration tools available, see 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 stores 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 1-4).

Table 1-4 WebCenter Portal Configuration Location

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

Discussions Server Configuration

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.

Oracle WebCenter Portal State and Configuration Persistence

WebCenter Portal runs as a J2EE application 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 Considerations

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.

Oracle WebCenter Portal Log File Locations

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

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

For example, diagnostics for WebCenter Portal are logged to: /base_domain/servers/WC_Portal/logs/WC_Portal-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 Viewing and Configuring Log Information.

Understanding the Oracle WebCenter Portal Installation

Installing WebCenter Portal 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 Oracle WebCenter Portal installation and post-installation administration tasks, see Roadmap for Installing and Configuring the Standard Installation Topologies in *Installing and Configuring Oracle WebCenter Portal*.

For post installation high availability configuration, see Setting up a Highly Available Environment in *High Availability Guide*.

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. Your ability to perform administration tasks depends on the Oracle WebLogic Server role you are assigned to —Admin, Operator, or Monitor. Table 1-5 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 1-5 WebCenter Portal Operations and Oracle WebLogic Server Roles

Operation	Admin Role	Operator Role	Monitor Role
Start and stop	Yes	Yes	No
View performance metrics	Yes	Yes	Yes

Table 1-5 (Cont.) WebCenter Portal Operations and Oracle WebLogic Server Roles

Operation	Admin Role	Operator Role	Monitor Role
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
Export entire application	Yes	No	No
Import entire application	Yes	No	No

Table 1-6 summarizes which tools you can use to perform various administrative operations relating to WebCenter Portal. Oracle WebCenter Portal Administration Tools describes the administrative tools.

Table 1-6 WebCenter Portal Operations and Administration Tools

Operation	Fusion Middleware Control	WLST Commands	WebLogic Server Admin Console	WebCenter Portal Admin
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
Configure security	Yes	Yes	Yes	No
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

Table 1-6 (Cont.) WebCenter Portal Operations and Administration Tools

Operation	Fusion Middleware Control	WLST Commands	WebLogic Server Admin Console	WebCenter Portal Admin
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

Performance Monitoring and Diagnostics

Performance monitoring helps administrators identify issues and performance bottlenecks in their environment. Monitoring WebCenter Portal Performance describes the range of performance metrics available for WebCenter Portal 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.

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:

- Managing WebCenter Portal Security
- Configuring the Identity Store
- Managing Users and Application Roles
- Configuring Single Sign-On
- Configuring SSL
- Configuring Web Services Security
- Configuring Security for Portlet Producers

Data Migration, Backup, and Recovery

Oracle WebCenter Portal stores data related to its configuration and content for the various feature areas 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 staging and production environments.

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

Oracle WebCenter Portal Administration Tools

Oracle WebCenter Portal offers the following administration tools:

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

Administrators should use these tools, rather than edit the configuration files, to perform administrative tasks. For help to decide which tool is best for you, see Configuration Tools.

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 domain (such as one containing Oracle WebCenter Portal).

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 Getting Started Using Oracle Enterprise Manager Fusion Middleware Control in *Administering Oracle Fusion Middleware*.

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

- 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, and portlet producers

Displaying Fusion Middleware Control Console

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

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 Displaying the Oracle WebLogic Server Administration Console in *Administering Oracle Fusion Middleware*.

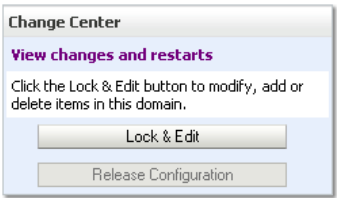
Locking Domain Configuration

You must lock configuration settings for a domain in the production mode before making any configuration changes. Navigate to the Administration Console's Change Center, and click **Lock & Edit**.

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

If the domain is in the development mode, the Lock & Edit option is not available, and changes are automatically committed.

Figure 1-3 Change Center in Oracle WebLogic Server Administration Console



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 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 fully administering and monitoring WebCenter Portal and managing connections to content repositories, portlet producers, external applications, and other back-end services. All Oracle WebCenter Portal WLST commands are described in WebCenterPortal Custom WLST Commands in *WebCenter WLST Command Reference Reference*.

Running Oracle WebLogic Scripting Tool (WLST) Commands

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

> **Note:**
>
> If you attempt to run WLST commands from the wrong directory, you will see a `NameError`. Always run the WLST commands from the Oracle home directory.
>
> See also, Troubleshooting WebCenter Portal.

To run WLST from the command line:

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

   (UNIX) *ORACLE_HOME*`/common/bin/wlst.sh`

   (Windows) *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',
   'protocol(optional):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
   - *protocol* is the protocol for connecting to the Administration Server and is optional
   - *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='t3://
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 *`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 is deployed to cluster.
- **Online documentation for Oracle WebCenter Portal WLST commands** is available in WebCenter Portal Custom WLST Commands in *WebCenter WLST Command Reference Reference*.

System MBean Browser

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

> **Note:**
>
> While you can monitor and configure WebCenter Portal MBeans from the System MBean browser, it is not the preferred tool for configuration. Oracle recommends that you configure WebCenter Portal settings from its home page using Fusion Middleware Control or by using WLST commands.

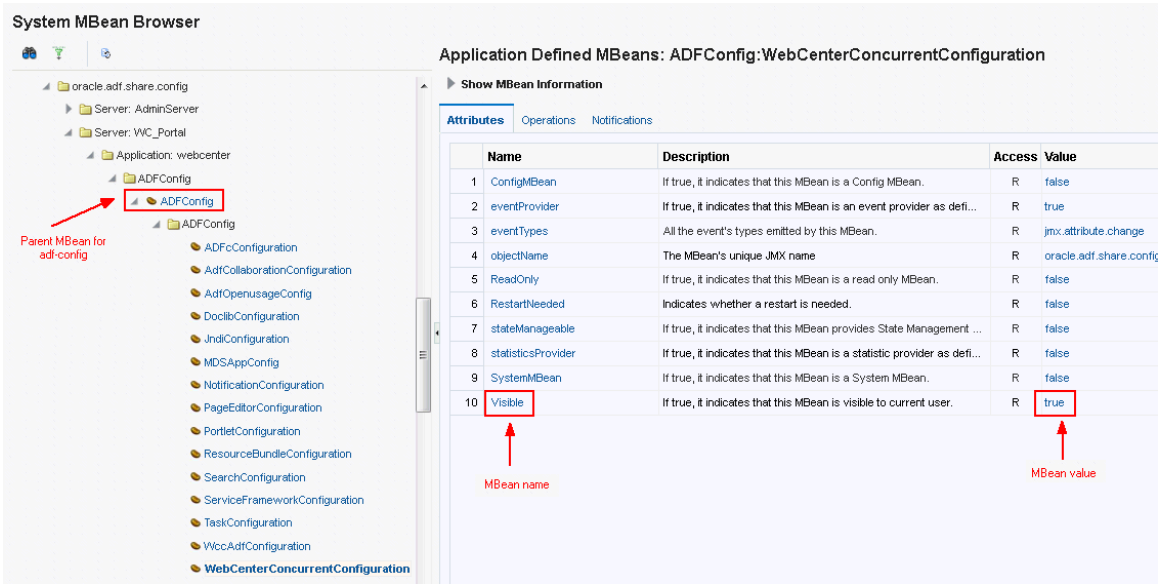
To access application MBeans:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal. For more information, see Navigating to the Home Page for WebCenter Portal.
2. From the **WebCenter Portal** menu, select **System MBean Browser**.
3. Under **Application Defined MBeans**, navigate to the MBean you want to view or configure.

   For example, you might want to navigate to MBeans for `adf-config.xml` and `connections.xml` as follows:

   - `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**
4. To view an MBean's attributes, select the MBean, then on the **Attributes** tab, click the required attribute. Values of some attributes can be changed. To do so, enter the value in the **Value** column.

Figure 1-4 Systems MBean Browser



5. Navigate to the parent MBean (for example, **ADFConfig** or **ADFConnections**), select the **Operations** tab, and click **save** to save the changes.
6. Restart the managed server on which WebCenter Portal is deployed. For more information, see Starting and Stopping Managed Servers for WebCenter Portal Application Deployments.

WebCenter Portal Administration Pages

WebCenter Portal provides several administration pages, which appear only to users who have logged in to WebCenter Portal using an administrator user name and password.

WebCenter Portal 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 information, see Accessing the Settings Pages in WebCenter Portal Administration.

Part II
Getting Started

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

- Getting Started Administering WebCenter Portal
- Starting Enterprise Manager Fusion Middleware Control
- Starting and Stopping Managed Servers and Applications for Oracle WebCenter Portal

2

Getting Started Administering WebCenter Portal

Before you get WebCenter Portal up and running, become familiar with the various administrative tasks you will perform as a Fusion Middleware administrator and as a WebCenter Portal administrator.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

> **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 WebCenter Portal Administration.
>
> Users with this role are also known as *WebCenter Portal administrators*.
>
> See also, Understanding Administrative Operations, Roles, and Tools.

Topics:

- Role of the System Administrator
- Installing WebCenter Portal
- Setting Up WebCenter Portal for the First Time (Roadmap)
- Customizing WebCenter Portal for the First Time (Roadmap)
- System Administration for WebCenter Portal – Fusion Middleware Admin Role (Roadmap)
- System Administration for WebCenter Portal – WebCenter Portal Admin Role (Roadmap)

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 ongoing 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 default system administrator account is created named `weblogic`. You can choose to create the account by any other name. 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 Server Administration Console or Oracle WebLogic Scripting Tool (WLST). For more information, see Administrative Users and Roles in *Securing Applications with Oracle Platform Security Services*.

  Oracle WebLogic Server provides two other roles, in addition to the `Admin` role, namely `Operator` and `Monitor`. For more information about these role, see Understanding Administrative Operations, Roles, and Tools.

- **WebCenter Portal Administration** - Assign one more users the `Administrator` role through WebCenter Portal Administration.

  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.

Installing WebCenter Portal

WebCenter Portal installation is described in Roadmap for Installing and Configuring the Standard Installation Topologies in *Installing and Configuring Oracle WebCenter Portal*.

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

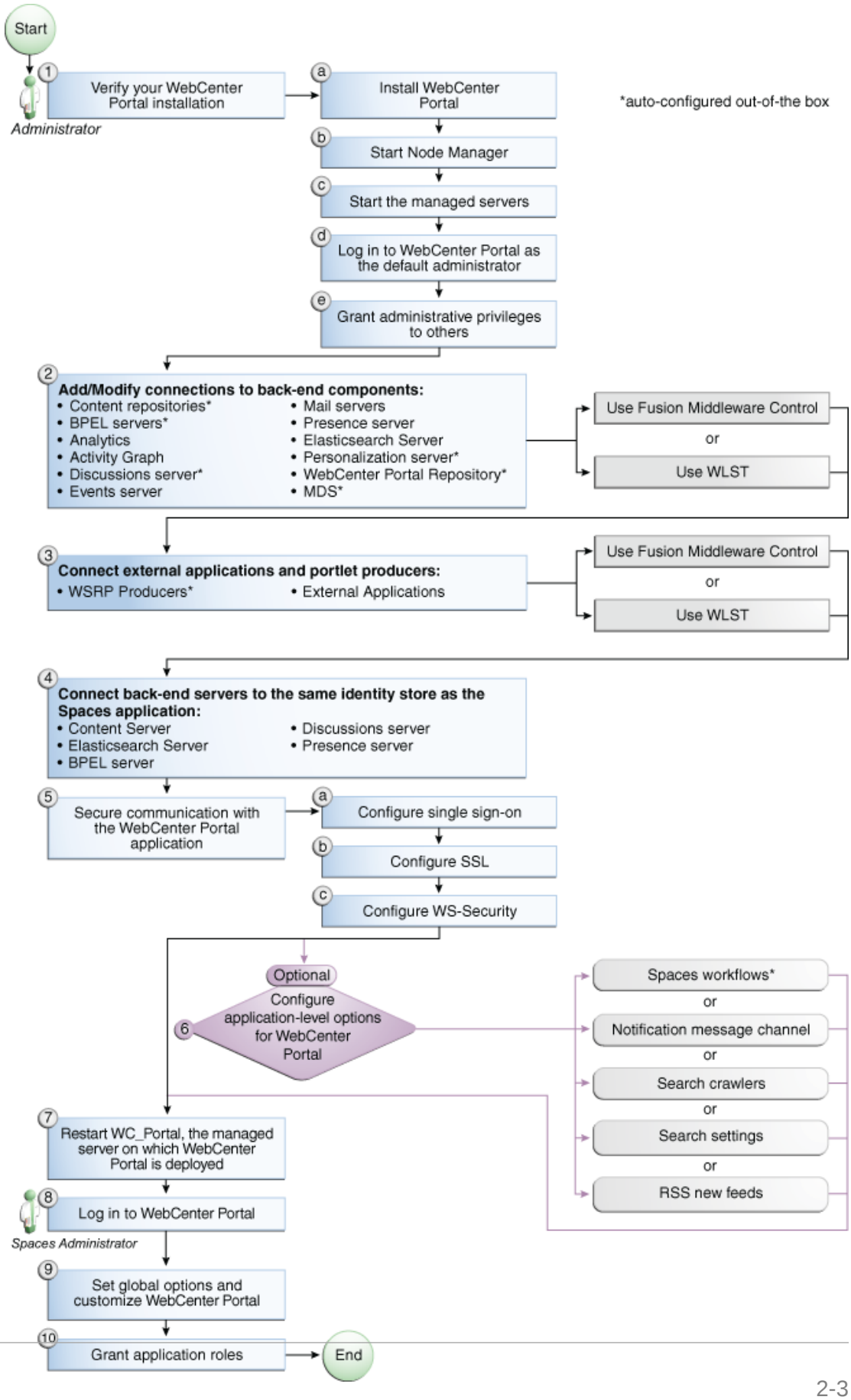


Table 2-1 Roadmap - Setting Up WebCenter Portal for the First Time

Actor	Task	Subtask	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** Grant Administartive Privileges	
Fusion Middleware Administrator	**2.** Add/modify connections to backend components using either of the following tools: • Fusion Middleware Control • WLST		Back-end components may include: • Content repositories[1] • BPEL servers[2] • Analytics collector • Events server • Mail servers • Presence server • Elasticsearch server • WebCenter Portal repository[3]
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[4]
Fusion Middleware Administrator	**4.** Connect back-end servers to the same identity store as WebCenter Portal.		Back-end servers may include: • Oracle WebCenter Content Server • Elasticsearch server • BPEL server • Discussions server • Presence server

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

Actor	Task	Subtask	Notes
Fusion Middleware Administrator	**5.** Secure communication with WebCenter Portal	**5.a** Configure single sign-on **5.b** Configure SSL **5.c** Configure WS-Security	Configuring SSO and SSL is optional.
Fusion Middleware Administrator	**6.** (Optional) Configure system options for WebCenter Portal: • Portal workflows[5] • Notification message channel • Configuring Search Crawlers • RSS news feeds		
Fusion Middleware Administrator	**7.** Restart WC_Portal, 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 Application Roles		

1 Auto-configured out-of-the-box
2 Auto-configured out-of-the-box
3 Auto-configured out-of-the-box
4 Auto-configured out-of-the-box
5 Auto-configured out-of-the-box

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 WebCenter Portal Administration Page **Tips:** WebCenter Portal URL is `http://host:port/webcenter` WebCenter Portal Administration URL is `http://host:port/webcenter/portal/admin/settings`	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: • Working with WebCenter Portal Administration Settings • 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 or the Self- Registering Page • Customizing the Login Page • Choosing a Default Display Language • Default Application Roles	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 WebCenter Portal Security • Assigning Users (and Groups) to Application Roles • Defining Application Roles • Assigning a User to a Different Application Role • Modifying Application Role Permissions	WebCenter Portal Admin
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

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

Task	Documentation	Actor
7. 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
8. Set up mail notifications	Configure default options for everyone's mail: • Configuring Send Mail Notifications	WebCenter Portal Admin
9. 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 portals they are likely to build: • Creating and Building a New Portal • Creating a New Portal Template	WebCenter Portal Admin

System Administration for WebCenter Portal – Fusion Middleware Admin Role (Roadmap)

The roadmap in Table 2-3 outlines typical tasks that a system administrator might perform to keep WebCenter Portal up and running.

Table 2-3 Roadmap - Administering and Monitoring WebCenter Portal

Task	Documentation	Role
Stop and start the managed servers	Restart the managed servers for configuration changes to take effect or for routine maintenance: • Starting and Stopping Managed Servers for WebCenter Portal Application Deployments Tip: The managed server for WebCenter Portal is named `WC_Portal`.	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
Monitor performance	Analyze the performance of the WebCenter Portal application and monitor its current status through Fusion Middleware Control: • Viewing Performance Metrics Using Fusion Middleware Control • Using Key Performance Metric Data to Analyze and Diagnose System Health System administrators granted one of these WebLogic Server roles can view performance metrics: `Admin`, `Operator`, `Monitor`. To find out more, see Understanding Administrative Operations, Roles, and Tools. WebCenter Portal administrators can monitor application performance and usage using WebCenter Portal's analytics feature: • Understanding the Analytics Administration Page in WebCenter Portal	Fusion Middleware Admin WebCenter Portal Admin

Table 2-3 (Cont.) Roadmap - Administering and Monitoring WebCenter Portal

Task	Documentation	Role
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.	Fusion Middleware Admin
• **Content repositories**	• Managing Connections to Oracle WebCenter Content Server	
• **Mail servers**	• Managing Mail	
• **BPEL servers**	• Managing the SOA Connection for WebCenter Portal Membership Workflows	
• **Collaboration**	• Managing Instant Messaging and Presence	
• **Calendar**	• Managing Calendar Events	
• **Elasticsearch**	• Configuring Search in WebCenter Portal	
• **Analytics**	• Managing Analytics	
• **Events, Links, Lists, Notes, Tags, and People Connections**	Managing the Metadata Repository in *Administering Oracle Fusion Middleware*.	
Modify external applications and portlet producers	Add, modify, and delete connections through Fusion Middleware Control.	Fusion Middleware Admin
• **External Applications**	• Managing External Applications	
• **Portlet Producers**	• Registering WSRP Producers	
Configure SSL communication	Configure secure communication: • Configuring SSL • Configuring Web Services Security • Configuring Single Sign-On	Fusion Middleware Admin
Reassociate your identity, policy, and credential stores	Reassociate your identity or policy stores: • Configuring the Identity Store • Managing Users and Application Roles	Fusion Middleware Admin
Reconfigure WebCenter Portal repository	Reconfigure the WebCenter Portal repository:	Fusion Middleware Admin
Reconfigure MDS repository	Reconfigure the application's MDS repository: • 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

Table 2-3 (Cont.) Roadmap - Administering and Monitoring WebCenter Portal

Task	Documentation	Role
Reconfigure WebCenter Portal workflows	Install WebCenter Portal workflows on a different BPEL server and reconfigure the connection: • 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	Fusion Middleware Admin
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	Fusion Middleware Admin

System Administration for WebCenter Portal – WebCenter Portal Admin Role (Roadmap)

The roadmap in Table 2-4 outlines typical tasks that a system administrator might perform while WebCenter Portal is up and running.

If WebCenter Portal 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 2-4 Roadmap - Keeping WebCenter Portal Up and Running

Task	Documentation	Role
Modify application Settings	Modify application-wide settings as required: • Working with WebCenter Portal Administration Settings • 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

Table 2-4 (Cont.) Roadmap - Keeping WebCenter Portal Up and Running

Task	Documentation	Role
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
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

3

Starting Enterprise Manager Fusion Middleware Control

Use Oracle Enterprise Manager Fusion Middleware Control Console to configure, monitor, and manage WebCenter Portal. Learn how to access the console and the home page for WebCenter Portal.

> **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, Understanding Administrative Operations, Roles, and Tools.

Topics:

- Displaying Fusion Middleware Control Console
- Navigating to the Home Page for WebCenter Portal
- Navigating to Dependent Components

Displaying Fusion Middleware Control Console

System administrators can log in to Fusion Middleware Control Console and access pages for managing Oracle WebCenter Portal. Fusion Middleware Control is usually automatically started when you start an Oracle WebLogic Server Administration Server. Your role determines what you can see and do after logging in.

To access the Fusion Middleware Control Console:

1. Start the Oracle WebLogic Server Administration Server using the WLST command line or a script.

   For example, use the following script:

   DOMAIN_HOME`/bin/startWebLogic.sh`

2. Enter the following URL in your 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`

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

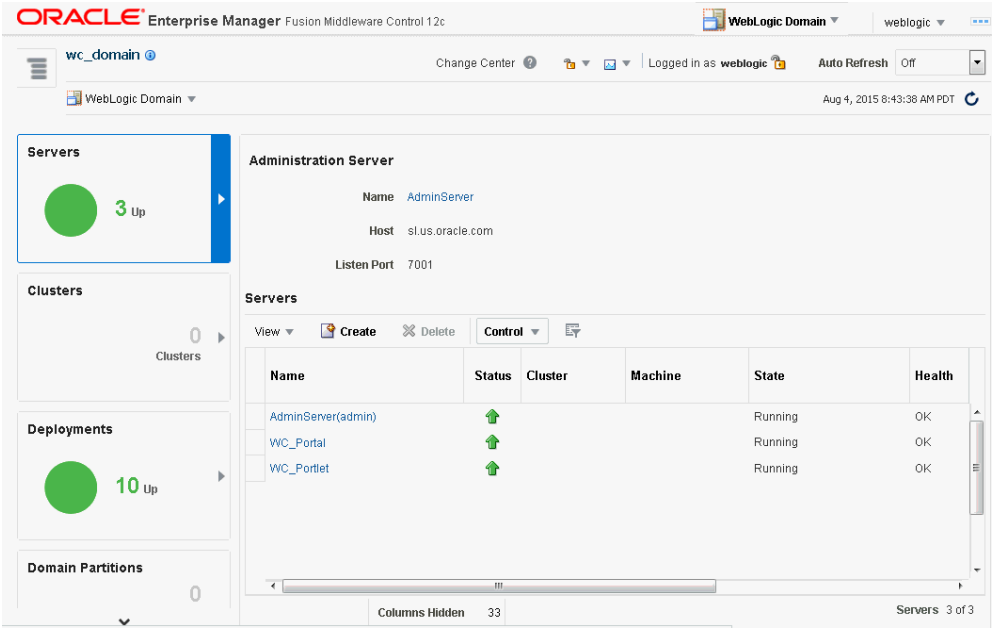
The default user name for the administrator user is `weblogic`. This is the account you can use to log in to Fusion Middleware Control for the first time.

4. Click **Login**.

The first page you see is the Domain home page (Figure 3-1). You can view this page at any time by selecting the name of the domain in the navigation pane.

> **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 3-1 Domain Home Page



From the navigation pane, you can drill down to view and manage all components in your domain, including WebCenter Portal.

Navigating to the Home Page for WebCenter Portal

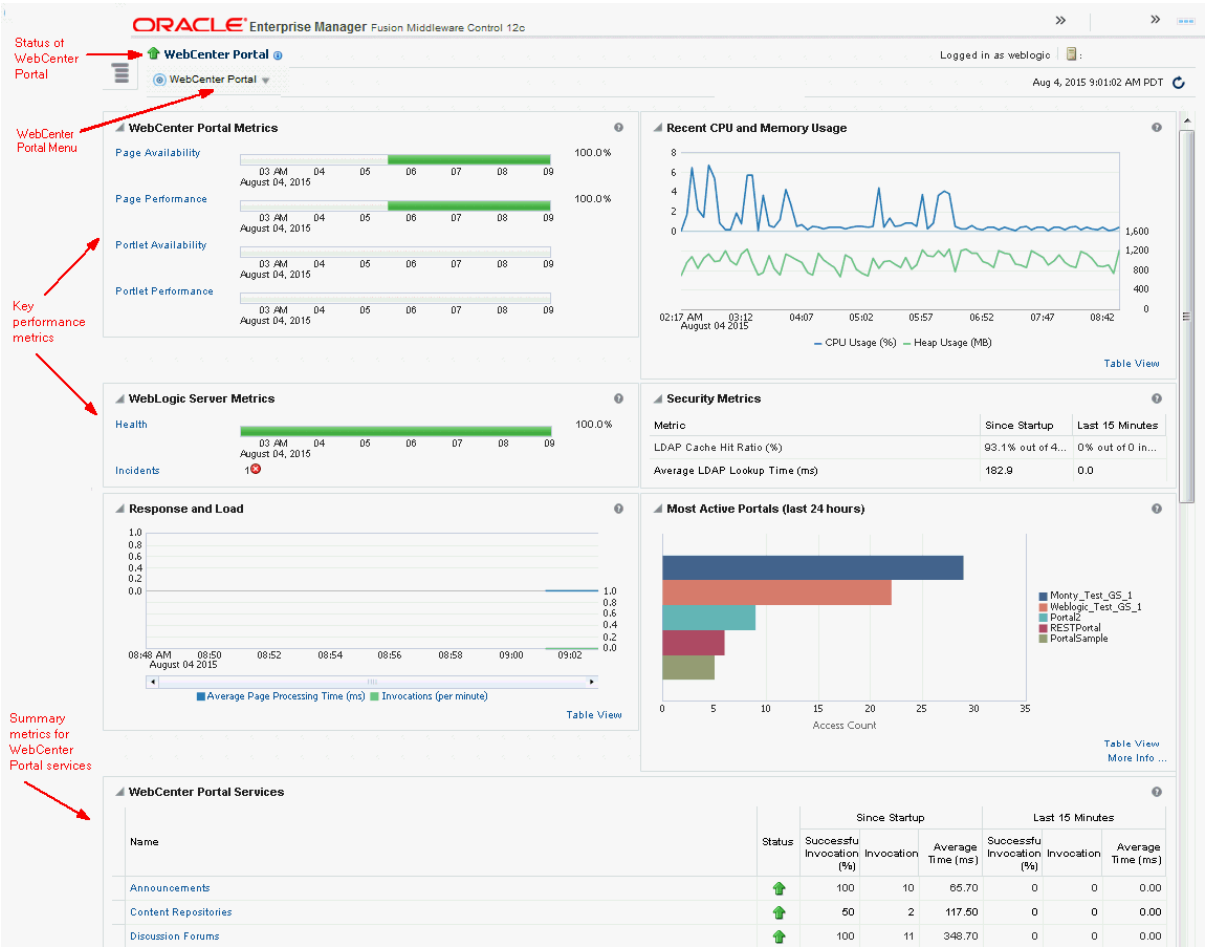
This section includes the following topics:

- Home Page for WebCenter Portal.
- Navigating to the WebCenter Portal Home Page.

Home Page for WebCenter Portal

The WebCenter Portal home page is your starting place for managing WebCenter Portal. The page displays status, performance and availability of all the components and tools or services that make up WebCenter Portal.

Figure 3-2 WebCenter Portal Home Page



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, 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 performs 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 Using Key Performance Metric Data to Analyze and Diagnose System Health .

The home page for WebCenter Portal also displays a **WebCenter Portal** menu.

Figure 3-3 Menu for the WebCenter Portal Application

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

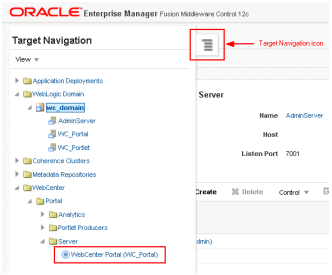
- Analyze diagnostic information and configure logs
- Export and import the WebCenter Portal application
- Register and manage portlet producers
- Configure application settings
- Manage back-end services
- Manage external applications
- Configure security policies and roles
- Configure ADF and MDS options
- View web services-related information

Navigating to the WebCenter Portal Home Page

To navigate to the main home page for WebCenter Portal:

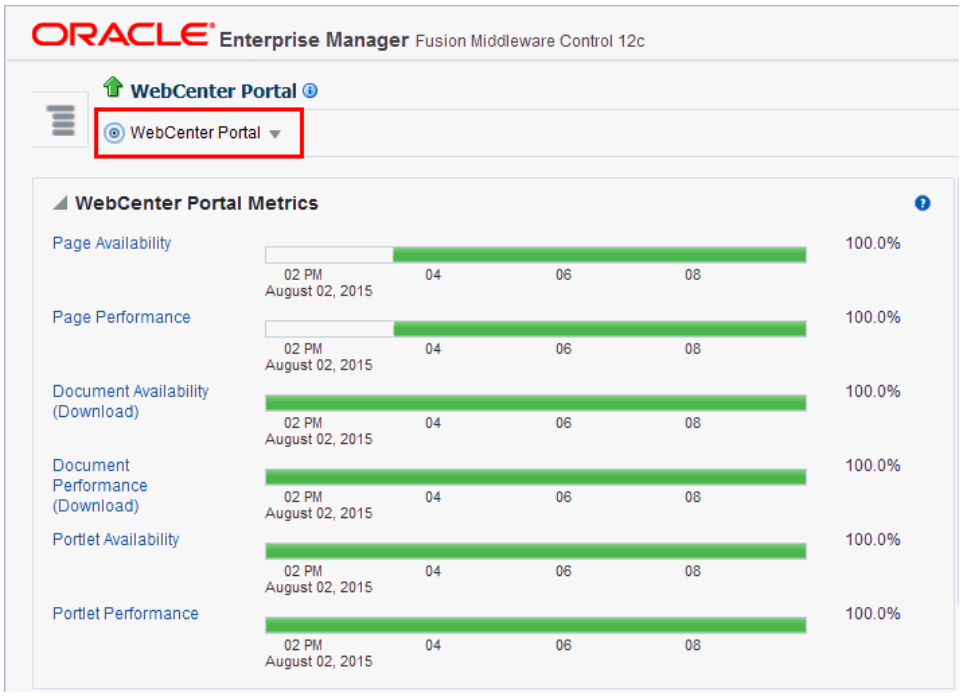
1. Log on to Fusion Middleware Control.
2. Click the **Target Navigation** icon at the top of the page.
3. In the **Target Navigation**, expand **WebCenter > Portal > Server**.
4. Select **WebCenter Portal (`WC_Portal`)** to navigate to the home page for your WebCenter Portal installation (Figure 3-4).

Figure 3-4 Navigating to the WebCenter Portal Home Page



Notice how the Navigator menu changes to **WebCenter Portal**.

Figure 3-5 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 (`WC_Portal`)** node (under the **Server** node on the left in Figure 3-4), the same *WebCenter Portal* menu displays.

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, and MDS repository.

On the WebCenter Portal home page, click the links in the **Related Components** section to navigate to WebCenter Portal application itself, WebLogic Server installation pages, or MDS repository pages in Fusion Middleware Control. See also, Navigating to the Home Page for WebCenter Portal.

4

Starting and Stopping Managed Servers and Applications for Oracle WebCenter Portal

Most configuration changes that you make to WebCenter Portal through Fusion Middleware Control or WLST commands are not dynamic. For changes to take effect, you must restart managed servers.

There are exceptions. Portlet producer and external application registrations are dynamic. Any new portlet producers and external applications that you register are immediately available in WebCenter Portal. Also, any changes to existing connections take effect immediately.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

This chapter includes the following topics:

- Starting Node Manager
- Starting and Stopping Managed Servers for WebCenter Portal Application Deployments
- Starting and Stopping the WebCenter Portal Application

> **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, Understanding Administrative Operations, Roles, and Tools.

> **Note:**
>
> You can perform all start and stop operations from the Oracle WebLogic Server Administration Console too. See Starting and Stopping Servers in *Administering Server Startup and Shutdown for Oracle WebLogic Server*.
>
> Node Manager *must* be running before you can start and stop administration servers, managed servers, and WebCenter Portal through Fusion Middleware Control or Oracle WebLogic Server Administration Console. Alternatively, you can start administration servers or managed servers from the command line using the `startWeblogic.sh` or `startManagedWebLogic.sh` scripts, respectively.

Starting Node Manager

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

For information on how to start Node Manager with `startNodeManager.sh`, see Using Node Manager in *Administering Node Manager for Oracle WebLogic Server*.

Starting and Stopping Managed Servers for WebCenter Portal Application Deployments

This section includes the following sections:

- Oracle WebCenter Portal Managed Servers
- Starting and Stopping Managed Servers

Oracle WebCenter Portal Managed Servers

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.

Table 4-1 Oracle WebCenter Portal Managed Servers and Applications

Managed Server	Application(s)
WC_Portal	webcenter (WebCenter Portal application) webcenter-help (WebCenter Portal Online Help) analytics-collector (Analytics)
WC_Portlet	wsrp-tools (WSRP Tools)
WC_Collaboration	owc_discussions (Discussions Server)

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 is deployed last.

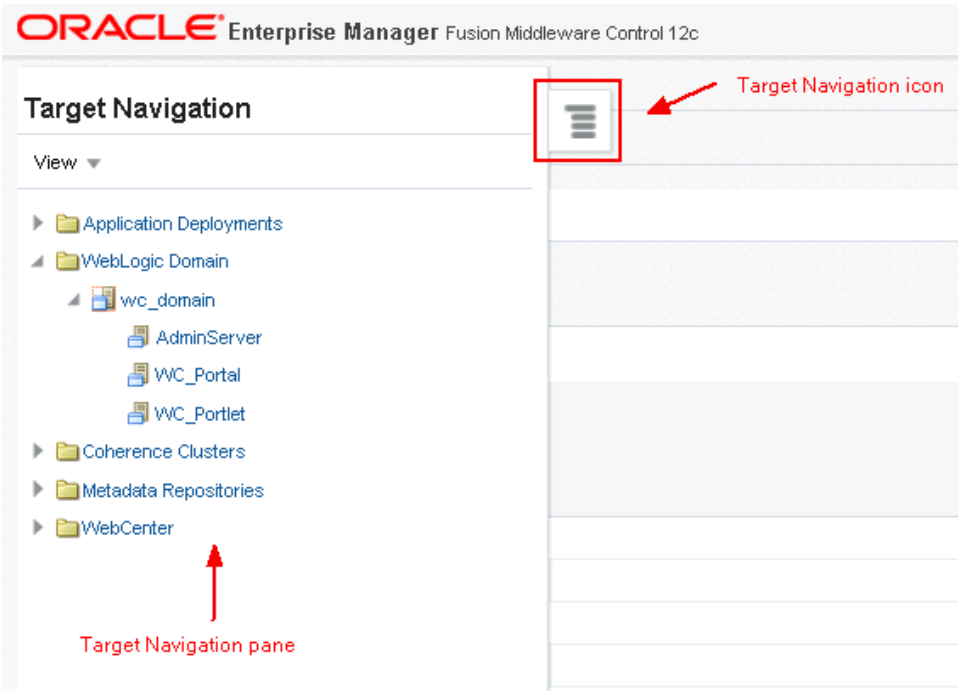
Starting and Stopping Managed Servers

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

1. Log in to Fusion Middleware Control.
2. Click the Target Navigation icon and expand **WebLogic Domain** in the Target Navigation pane.
3. Expand **wc_domain**, and select the managed server you want to start or stop (Figure 4-2).

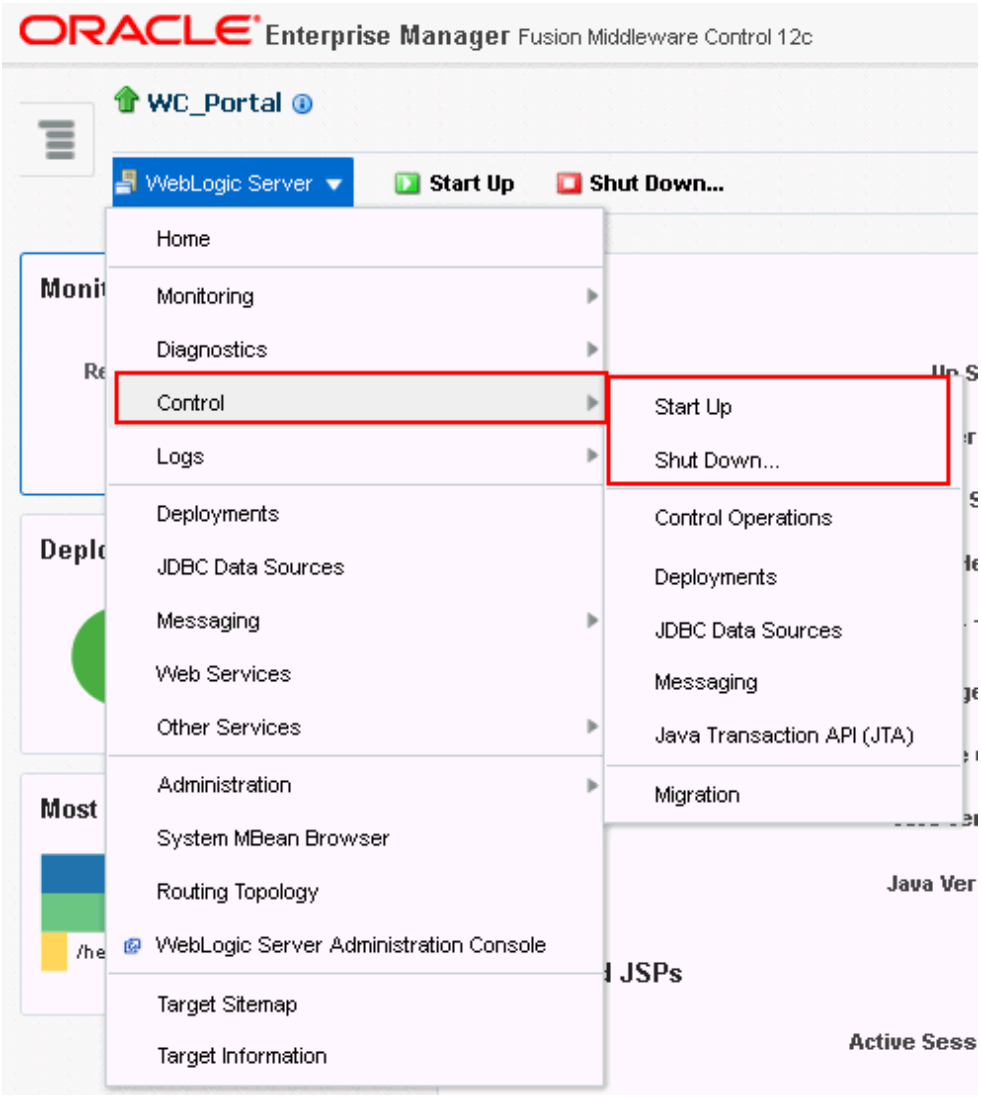
   The home page for the managed server displays.

Figure 4-1 Accessing 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 4-2 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 Starting and Stopping Oracle WebLogic Server Instances in *Administering Oracle Fusion Middleware*.

Starting and Stopping the WebCenter Portal Application

You can start, restart, and shut down WebCenter Portal by using Fusion Middleware Control or WLST.

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

- Stopping WebCenter Portal Using WLST

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

   Alternatively, right-click **WC_Portal** 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).

Starting WebCenter Portal Using WLST

Use the WLST command `startApplication` to start WebCenter Portal. For command syntax and detailed examples, see startApplication in *WLST Command Reference for WebLogic Server*.

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

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

   Alternatively, right-click **WC_Portal** 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).

Stopping WebCenter Portal Using WLST

Use the WLST command `stopApplication` to stop the WebCenter Portal application. For command syntax and detailed examples, see stopApplication in *WLST Command Reference for WebLogic Server*.

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

Part III
Administering Tools and Services

This part of *Administering Oracle WebCenter Portal* provides information about the administration tasks for tools, services, portlet producers, and external applications used by Oracle WebCenter Portal.

- Managing Tools and Services
- Managing Connections to Oracle WebCenter Content Server
- Managing Analytics
- Managing Calendar Events
- Integrating Other Oracle Applications
- Managing Instant Messaging and Presence
- Managing Mail
- Managing People Connections
- Managing RSS
- Configuring Search in WebCenter Portal
- Managing Subscriptions and Notifications
- Managing the SOA Connection for WebCenter Portal Membership Workflows
- Managing Portlet Producers
- Managing External Applications
- Managing REST Services

5

Managing Tools and Services

WebCenter Portal supports tools and services that expose collaborative, social networking, and personal productivity features in portals. While certain features are available by default, for other features you need to install and configure additional back-end servers like WebCenter Content.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

> **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 WebCenter Portal Administration.
>
> For more information about roles and permissions, see Understanding Administrative Operations, Roles, and Tools.
>
> The tasks described 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 manager task, as described in the Introduction to Portal Tools and Services in *Building Portals with Oracle WebCenter Portal*.

Topics:

- About Tools and Services
- Enabling and Disabling Tools and Services in WebCenter Portal

About Tools and Services

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 personal notes, are ready to use out-of-the-box and require no further configuration. Other tools, such as documents, and other services, such as mail, require connections to the back-end server and require additional configuration.

Tools and services available out-of-the-box

The following table lists the tools and services available out-of-the-box in WebCenter Portal:

Table 5-1 Tools and services available out-of-the-box

Tools and Services	Description	External Repository	For More Information
Analytics	Enables you to display usage and performance metrics for your portal application		Managing Analytics
Events	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.		Group events are available out-of-the-box and Personal events need additional configuration. Managing Calendar Events
Links	Provides the ability to view, access, and associate related information; for example, you can link to a document from an event		Linking Information in WebCenter Portal in *Using Portals in Oracle WebCenter Portal*
Lists	Provides the ability to create, publish, and manage lists		Adding Lists of Information to a Portal in *Building Portals with Oracle WebCenter Portal*
Notes	Provides the ability to "jot down" and retain bits of personally relevant 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		Managing Subscriptions and Notifications

Table 5-1 (Cont.) Tools and services available out-of-the-box

Tools and Services	Description	External Repository	For More Information
People Connections	Provides social networking capabilities, such as creating a personal profile, displaying current status, and viewing other users' activities. You can set options for people connection features. People Connections 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 information about yourself and viewing the information of other users • Publisher for publishing status messages and posting files and links		Configuring People Connections for WebCenter Portal **Note**: People Connection is available out-of-the-box and need additional configuration. See Configuring People Connections for WebCenter Portal

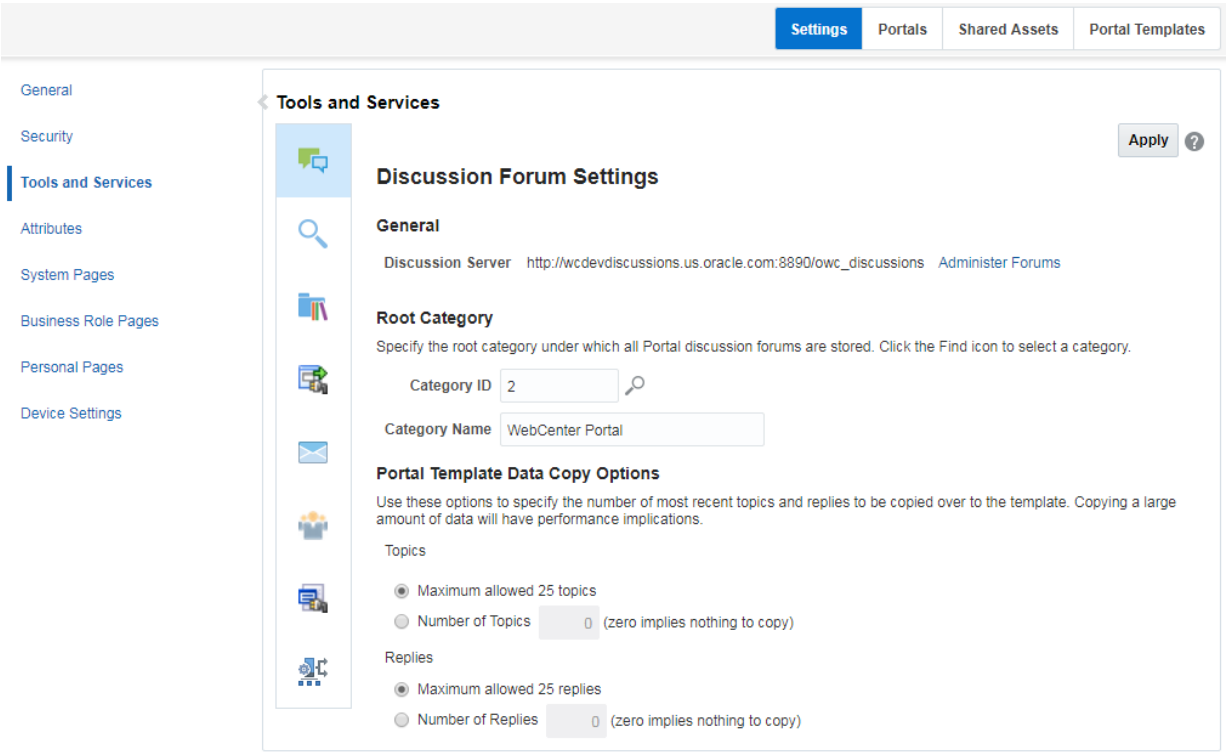
Table 5-1 (Cont.) Tools and services available out-of-the-box

Tools and Services	Description	External Repository	For More Information
RSS	Provides the ability to access the content of many different web sites from a single location—a news reader		Managing RSS
Tags	Provides the ability to assign one or more personally-relevant keywords to a given page		

Tools and services available with additional configuration

Some Tools and services become available in WebCenter Portalwhen you configure connections to the appropriate back-end applications. Portal managers are responsible for managing tools and services in their individual portals. As the system administrator, you can set up some additional configurations in the WebCenter Portal Administration Tools and Services page.

Figure 5-1 WebCenter Portal Tools and Services Page



The following table lists the tools and services that you can configure in WebCenter Portal Cloud Administration Tools and Services page.

Table 5-2 Tools and Services Configuration in WebCenter Portal Administration

Tool or Service	Description	External Repository	More Information
Events	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.		Group events are available out-of-the-box and Personal events need additional configuration. Managing Calendar Events
Documents	Provides content management and storage capabilities, including file upload, file and folder creation and management, file check out, versioning, and so on. The documents tool also supports the wiki and blog functionality.	Oracle WebCenter Content	Managing Connections to Oracle WebCenter Content Server Understanding the Content Manager Interface
External Applications	Provides an option register new external applications, or edit and deregister the existing external applications.		Managing External Applications at Runtime.
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	Microsoft Exchange server	Managing Instant Messaging and Presence Using Instant Messaging and Presence Viewer in *Using Portals in Oracle WebCenter Portal*
Mail	Provides access to mail services and ability to perform basic operations such as read, reply, and forward within WebCenter Portal. Specify the default mail client for either the local mail client or WebCenter Portal's mail service.	Microsoft Exchange Server	Configuring Send Mail Notifications for WebCenter Portal.

Table 5-2 (Cont.) Tools and Services Configuration in WebCenter Portal Administration

Tool or Service	Description	External Repository	More Information
People Connections	Provides social networking capabilities, such as creating a personal profile, displaying current status, and viewing other users' activities. You can set options for people connection features. People Connections 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 information about yourself and viewing the information of other users • Publisher for publishing status messages and posting files and links		Configuring People Connections for WebCenter Portal
Portlet Producers	Provides option to register new portlet producers, or edit and deregister existing portlet producers. .		Managing Portlet Producers
Portal Server Connections	Provides option to register new portal servers for deploying WebCenter Portal.		Creating a Portal Server Connection.

Table 5-2 (Cont.) Tools and Services Configuration in WebCenter Portal Administration

Tool or Service	Description	External Repository	More Information
Search	Provides the ability to search services, the application, or an entire site. You can customize your search setting based on the need, change the search scope, and configure the filtering options for advanced search.		Configuring Search in WebCenter Portal

Enabling and Disabling Tools and Services in WebCenter Portal

You, as a system administrator, are responsible for managing connections to external servers and maintain the database schema and Metadata Service (MDS) repositories where application data, specific to WebCenter Portal Cloud, is stored.

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 managers 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. Portal Managers can manually disable a tool in the portal, with the exception of the Mail service.

If a portal manager manually enables a tool in a portal, WebCenter Portal handles any necessary configuration with the back-end server. For example, when the portal manager 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. If role-mapping fails, the portal manager 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.

When a back-end server is not configured, intentionally or otherwise, WebCenter Portal Cloud 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 portal managers—through the portal's administration settings.

If a tool or service is temporarily unavailable, you can use Fusion Middleware Control to investigate, diagnose, and solve issues relating to services. 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.

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

- Announcement
- Discussion
- Events
- Lists
- Documents

> **Note:**
>
> In most cases, the portal managers 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 Enabling and Disabling Tools and Services Available to a Portal in *Building Portals with Oracle WebCenter Portal*.

6

Managing Connections to Oracle WebCenter Content Server

You can create connections to Oracle WebCenter Content Server to enable content integration within Oracle WebCenter Portal.
This chapter includes the following topics:

- About Oracle WebCenter Content Server Connections
- Prerequisites for Configuring Oracle WebCenter Content Server
- Configuration Roadmap for Oracle WebCenter Content Server
- Configuring Oracle WebCenter Content Server
- Creating a Connection to Oracle WebCenter Content Server
- Setting Connection Properties for the Default Oracle WebCenter Content Server Connection
- Modifying Oracle WebCenter Content Server Connection Details
- Deleting Oracle WebCenter Content Server Connections
- Changing the Maximum File Upload Size
- Configuring Content Manager for Oracle Content and Experience Cloud

> **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 WebCenter Portal granted through WebCenter Portal Administration.
>
> For more information about roles and permissions, see Understanding Administrative Operations, Roles, and Tools.

About Oracle WebCenter Content Server Connections

By leveraging the functionality of Oracle WebCenter Content Server, Oracle WebCenter Portal provides content management and storage capabilities, including content upload, file and folder creation and management, file check out, versioning, and so on.

To provide content integration in WebCenter Portal, you must configure at least one WebCenter Content Server connection and mark it as the *default* connection (sometimes referred to as the *active* or *primary* connection). Before creating a connection to WebCenter Content Server, you must complete the required prerequisites.

> **Note:**
>
> WebCenter Portal supports multiple Content Server connections.
>
> However, iFraming is supported only for the default Content Server connection. Therefore, when portal managers set properties for the Content Manager task flow or Content Presenter, they cannot specify a non-default Content Server connection if these task flows will use iFrames to display file content, such as PDF files.

> **Note:**
>
> It is recommended not to change the default Oracle WebCenter Content Server connection after it has been created as this may lead to unpredictable issues. **If you encounter any issues, contact Oracle Support**.

WebCenter Portal provides content integration through:

- **Content Manager task flow**, which enable users to view and manage documents and other types of content in WebCenter Content Server.
- **Content Presenter task flow**, which enables end users to select content from WebCenter Content Server in a variety of ways and then display those items using available display templates.
- **Wiki and Blog pages**, which enable users to create collaborative portal pages.
- **Content Contribution and Publishing**, which enables end users to add text, images, and video to portal pages. A connection to WebCenter Content Server is not required for content contribution and publishing, however if a WebCenter Content Server connection does exist, images that are stored in WebCenter Content Server can be published in Image components and links to WebCenter Content Server items can be added to Image and Text components.

Any portal (including the Home portal) that enables content integration has its own document folder in the WebCenter Content Server repository identified by WebCenter Portal's default WebCenter Content Server connection.

The content repository identified by the default WebCenter Content Server connection must be connected to the same identity store that is used by WebCenter Portal.

Just like other service connections, post-deployment WebCenter Content Server connections are registered and managed through Oracle Enterprise Manager Fusion Middleware Control or using the WLST command-line tool. Connection information is stored in configuration files and in the Oracle Metadata Services Repository.

Always use Fusion Middleware Control or the WLST command-line tool to review and configure back-end services for WebCenter Portal. All changes that you make, post deployment, are stored in the MDS Repository as customizations.

> **Note:**
>
> WebCenter Content Server connection 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 is deployed for your changes to take effect.

Once connection details are defined, users can expose the content of the connected WebCenter Content Server repository through the Content Manager and Content Presenter task flows. For more information, see Working with Content in a Portal in *Building Portals with Oracle WebCenter Portal* and Adding Content to a Portal in *Using Portals in Oracle WebCenter Portal*.

> **Note:**
>
> Although Microsoft SharePoint is not directly supported as a back-end content store for WebCenter Portal, it is possible to use Content Server as a repository for SharePoint documents. For more information, see Introduction in *Administering the Oracle WebCenter Content Storage Connector for Microsoft SharePoint*.

Prerequisites for Configuring Oracle WebCenter Content Server

Read this section to understand the prerequisites and other considerations before continuing with Oracle WebCenter Content Server.

This section includes the following topics:

- Installation Prerequisites for Oracle WebCenter Content Server
- Installation Prerequisites for Inbound Refinery
- Configuration Prerequisites for Oracle WebCenter Content Server and Inbound Refinery
- Security Prerequisites for Oracle WebCenter Content Server and Inbound Refinery

Installation Prerequisites for Oracle WebCenter Content Server

Oracle WebCenter Content Server is installed as part of Oracle WebCenter Content, which is an Oracle Fusion Middleware component.

For more information about installing WebCenter Content, see Installing Oracle WebCenter Content in *Installing and Configuring Oracle WebCenter Content*.

If you already have an earlier version of WebCenter Content Server installed, upgrade your installation to WebCenter Content 12*c* prior to configuring it. For information about upgrading to WebCenter Content 12*c*, see Upgrading Your Oracle WebCenter Content Environment in *Upgrading Oracle WebCenter*.

Installation Prerequisites for Inbound Refinery

For content integration in Oracle WebCenter Portal, it is recommended that you also install Oracle WebCenter Content: Inbound Refinery as part of the installation of WebCenter Content.

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 WebCenter Content Server. Installing Inbound Refinery is described in Configuring Inbound Refinery Settings (Single Node) in *Installing and Configuring Oracle WebCenter Content*.

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

Configuration Prerequisites for Oracle WebCenter Content Server and Inbound Refinery

After installing Oracle WebCenter Content Server and Inbound Refinery, you should configure the initial post-intallation settings, including additional Oracle WebCenter Portal-specific instructions.

General post-installation settings are described in Configuring the Content Server Instance in *Installing and Configuring Oracle WebCenter Content*. WebCenter Portal-specific instructions are provided in the tables below. Be sure to restart the servers after updating the settings.

Table 6-1 Configuration Prerequisites - WebCenter Content Server

Setting	Description
Server Socket Port	This is the intradoc port that WebCenter Portal connects to using RIDC (defaults to `4444`). This value is stored in the *`MW_HOME`*`/user_projects/domains/ucm_domain/ucm/ibr/config/config.cfg` configuration file for the WebCenter Content Server managed server as `IntradocServerPort`.
Incoming Socket Connection Address Security Filter	Server filter specifying which machines can access WebCenter Content Server through a socket connection. This value is stored in the configuration file for the managed server as `SocketHostAddressSecurityFilter`.

Table 6-1 (Cont.) Configuration Prerequisites - WebCenter Content Server

Setting	Description
Full Text Search (Optional, but strongly recommended)	Specifies the full-text search engine. `SearchIndexerEngineName=ORACLETEXTSEARCH` is the recommended value.

Table 6-2 Configuration Prerequisites - Inbound Refinery

Setting	Description
Server Socket Port	This port is used for communication between WebCenter 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 which machines can access Inbound Refinery through RIDC. This value is stored in the configuration file for the managed server as `SocketHostAddressSecurityFilter`.

Security Prerequisites for Oracle WebCenter Content Server and Inbound Refinery

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

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

Oracle also recommends that you install and configure a single sign-on solution to avoid users having to log in twice when accessing WebCenter Content Server and other WebCenter Portal components. For more information about single sign-on solutions, see Configuring Single Sign-On.

Configuration Roadmap for Oracle WebCenter Content Server

To provide content integration, you must configure Oracle WebCenter Content Server to work with Oracle WebCenter Portal.

Table 6-3 provides an overview of the tasks required to configure Content Server for WebCenter Portal.

Table 6-3 WebCenter Portal Configuration Tasks for WebCenter Content Server

Task	Description	Documentation
Enable the mandatory components	**Mandatory** You must enable the WebCenterConfigure component (which configures an instance of WebCenter Content Server for WebCenter Portal). You must also enable the FrameworkFolders component (which provides a hierarchical folder interface to content in WebCenter Content Server). Also enable the DynamicConverter component. This provides a transformation technology and on-demand publishing solution for content items.	See Enabling Mandatory Components.
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 WebCenter Content Server.	See Configuring the Inbound Refinery.
Configure Secure Sockets Layer (SSL) for WebCenter Content Server	**Optional, but strongly recommended** To ensure secure identity propagation, you should set up SSL for WebCenter Content Server.	See Setting Up SSL for Oracle WebCenter Content Server.
Configure the SES Crawler	**Optional** You can override the default search adapters and use Elasticsearch to get unified ranking results for WebCenter Portal resources such as documents, pages, people, and so on.	See Configuring WebCenter Content for Search

Table 6-3 (Cont.) WebCenter Portal Configuration Tasks for WebCenter Content Server

Task	Description	Documentation
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, Oracle strongly recommends installing and configuring it so you don't have to come back to it later.	See Setting Up Site Studio. See also Oracle Site Studio Software Suite in *Managing Oracle Site Studio*.
Enable a Full-Text Search Option	**Optional, but strongly recommended** Although configuring full-text searching and indexing capabilities is nominally optional, Oracle strongly recommends that you 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 Enabling Full-Text Search.
Create Content Profiles	**Optional** Users have the option to upload content based on Content Profiles	See Creating Content Profiles in Oracle WebCenter Content Server.
Configure Access Control List Security	**Optional** For additional security of your content, you can configure Content Server to support access control lists (ACLs), which are lists of users, groups, or enterprise roles with permission to access or interact with a content item.	See Configuring Access Control List Security.
Enable Digital Asset Manager	**Optional** If you want to use Content Presenter to use different renditions of images in your portal, you may want to enable Digital Asset Manager (DAM) in WebCenter Content Server.	See Enabling Digital Asset Manager.

Table 6-3 (Cont.) WebCenter Portal Configuration Tasks for WebCenter Content Server

Task	Description	Documentation
Additional Optional Configurations	**Optional** After completing the rest of your configuration, you can optionally configure desktop integration, configure the FileStore Provider component, and set up Node Manager.	See Additional Optional Configurations for Oracle WebCenter Content Server.
Register Content Server	**Mandatory** 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.	See Configuring the Default Oracle WebCenter Content Server Connection for Oracle WebCenter Portal.
Enable Annotations for WebCenter Content Server	**Optional** To work with annotations in WebCenter Content, you need to have one of the following permissions: • Standard Annotation (S) • Restricted Annotation (T) • Hidden Annotation (H)	See About Permissions in *Administering Oracle WebCenter Content*

Creating a Connection to Oracle WebCenter Content Server

A default connection to Oracle WebCenter Content Server may be automatically created and configured when Oracle WebCenter Portal first starts up, but you may want to change the default settings or register other WebCenter Content Server repositories.

This section contains the following topics:

- About Creating a Connection to Oracle WebCenter Content Server
- Creating a Connection to Oracle WebCenter Content Server Using Fusion Middleware Control
- Registering Oracle WebCenter Content Server Using WLST
- Oracle WebCenter Content Server Connection Parameters for RIDC Socket Types

About Creating a Connection to Oracle WebCenter Content Server

When creating a connection to Oracle WebCenter Content Server, there are several things to think about.

Consider the following:

- Oracle WebCenter Portal communicates with WebCenter Content Server over the Remote Intradoc Client (RIDC). RIDC provides the ability for WebCenter Portal to remotely execute WebCenter Content Server services. It also handles things like connection pooling, security, and protocol specifics. RIDC supports socket-based communication and the HTTP and JAX-WS protocols.
  - **Socket** — The socket protocol communicates to WebCenter Content Server over the Intradoc socket port. This protocol requires a trusted connection between WebCenter Portal and WebCenter Content Server and does not perform any password validation.

    The socket-based communication can also be configured to run over SSL to provide extra security
  - **HTTP** — Using the HTTPClient package, RIDC communicates with the web server attached to WebCenter Content Server. Unlike the socket protocol, this protocol requires authentication credentials for each request.

    The HTTP protocol can also be load-balanced using an HTTP load balancer such as Oracle Traffic Director.
  - **JAX-WS** — With JAX-WS, WebCenter Portal must authenticate with WebCenter Content Server for each connection rather than assuming any connection from WebCenter Portal is automatically a trusted connection. The JAX-WS protocol is only supported in Oracle WebCenter Content with a properly configured WebCenter Content Server instance and the RIDC client installed.
- Additional configuration is required for the default WebCenter Content Server connection:
  - A user name with administrative rights for the WebCenter Content Server instance is required (**Content Administrator**). This user is used to create and maintain folders for portal content, security groups and roles, and manage content access rights. The content administrator is `wccadmin`. See Creating a Custom Administrator .

    Administrative privileges are required for this connection so that operations can be performed on behalf of WebCenter Portal users.
  - The **Portal Server Identifier** value is used as the name for the root folder within the WebCenter Content Server repository under which all WebCenter Portal content is stored. For the **Portal Server Identifier** value, you must specify a WebCenter Content Server folder that does not yet exist. Use the format: `/foldername`. For example: `/MyWebCenterPortal`. The **Portal Server Identifier** cannot be `/`, the WebCenter Content Server root itself, and it must be unique across different portals. The folder specified is created for you when WebCenter Portal starts up. Invalid entries include: `/`, `/foldername/`, `/foldername/subfolder`.
  - The **Security Group**, identifies a WebCenter Portal instance within this WebCenter Content Server 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 **Security Group** value is used for the following:
    - To separate data when multiple WebCenter Portal instances share the same WebCenter Content Server instance and should be unique across applications.
    - 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 *`securityGroup`*`User` and *`securityGroup`*`AuthenUser`).

    - 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 Managing Security Groups, Roles, and Permissions in *Administering Oracle WebCenter Content*. For information about folders, see Organizing Content in *Managing Oracle WebCenter Content*.

- **Portal Server Identifier** and **Security Group** values:
  - For the default connection in WebCenter Portal, the **Portal Server Identifier** and **Security Group** values are used to create the seed data in WebCenter Content Server to enable storage of portal-related data.

    > **WARNING:**
    >
    > You should never change the **Portal Server Identifier** or **Security Group** values separately; you should always change both. That is, if you change the **Portal Server Identifier** value after configuring and running WebCenter Portal, then you must also change the **Security Group** value, and vice versa. That is, you must change both values (**Portal Server Identifier** and **Security Group**) to unique values if WebCenter Portal already contains the seed data.

    When you change these values, the existing seed data is not renamed in WebCenter Content Server. 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 **Portal Server Identifier** folder and existing data under the old 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 **Portal Server Identifier**.

    > **Note:**
    >
    > Although the **Portal Server Identifier** and **Security Group** values change, the old folder still appears in search results, like any other folder in WebCenter Content Server.

- At start up, WebCenter Portal creates seed data (if it does not already exist) in the default WebCenter Content Server repository for WebCenter Portal.

Creating a Connection to Oracle WebCenter Content Server Using Fusion Middleware Control

You can register Oracle WebCenter Content Server as a content repository for Oracle WebCenter Portal using Fusion Middleware Control.

This section includes the following topics:

- Connecting to Oracle WebCenter Content Server Using Socket-Based Communication
- Connecting to Oracle WebCenter Content Server Using Secure Socket-Based Communication
- Connecting to Oracle WebCenter Content Server Using JAX-WS
- Connecting to Oracle WebCenter Content Server Using HTTP

Connecting to Oracle WebCenter Content Server Using Socket-Based Communication

The socket protocol communicates to Oracle WebCenter Content Server over the Intradoc socket port.

To connect to WebCenter Content Server using socket-based communication:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal.
2. From the WebCenter Portal menu, select **Settings** and then **Service Configuration**.
3. On the WebCenter Portal Services Configuration page, from the list of services select **Content Repository**.
4. Click **Add**.
5. In the **Connection Name** field, enter a unique name for the WebCenter Content Server connection.

   > **Tip:**
   >
   > The name must be unique (across all connection types) within WebCenter Portal.

6. The **Repository Type** is **Oracle Content Server**.
7. Select **Active Connection** to make this the default WebCenter Content Server connection for WebCenter Portal.

   You can create connections to multiple WebCenter Content Server instances; all connections are used. One connection must be the default connection. The default connection is the one used by WebCenter Portal to store portal-related documents.

   If this is the default connection for WebCenter Portal, some additional configuration is required. For more information, see Setting Connection Properties for the Default Oracle WebCenter Content Server Connection Using Fusion Middleware Control.

   > **Note:**
   >
   > Deselecting this option does not disable the connection. If a connection is no longer required, you must delete the connection.

8. From the **RIDC Socket Type** dropdown list, select **Socket**.
9. In the **Server Host** field, enter the host name of the machine where WebCenter Content Server is running.

For example: `mycontentserver.example.com`.

10. In the **Server Port** field, enter the port specified for the WebCenter Content Server's `incoming` provider.

    This property corresponds to the `IntradocServerPort` setting in the WebCenter Content Server configuration file, which defaults to `4444`.

    > **Tip:**
    >
    > You can find the current value by logging into WebCenter Content Server and navigating to **Administration > Admin Server > General Configuration > Additional Configuration Variables > IntradocServerPort**.

11. In the **Connection Timeout** field, enter the length of time (in milliseconds) to attempt to log in to WebCenter Content Server before issuing a connection timeout message.

    This value is also used as the socket timeout for the underlying RIDC connection for all service requests.

    If the **Connection Timeout** is not set, the following values are used:

    - **Login timeout** — the default concurrency timeout configured for the `oracle.webcenter.content` resource (30 seconds or 30000 milliseconds).
    - **RIDC socket timeout** — the default RIDC socket timeout (60 seconds or 60000 milliseconds).

    > **Tip:**
    >
    > It is recommended that you do not specify a value less than 60000 milliseconds for the **Connection Timeout**, 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.

12. From the **Authentication Method** dropdown list, select:

    - **Identity Propagation** if WebCenter Content Server uses the same identity store as WebCenter Portal to authenticate users.

      If you select this option, you must also specify the appropriate **Web Server context root for Content Server**.

    - **External Application** if WebCenter Content Server uses an external application to authenticate users. Select this option if you want to use public, shared, or mapped credentials.

      If you select this option, you must also specify the appropriate **Associated External Application**.

13. (Only if **Authentication Method** is **Identity Propagation**) In the **Web Server context root for Content Server** field, enter the web server context root for WebCenter Content Server if WebCenter Content Server is front-ended with Oracle HTTP Server (OHS).

Use the format `/contextRoot`. For example, `/cs`.

Oracle recommends that you access WebCenter Portal through Oracle HTTP Server (OHS) if you want to use Content Presenter to create or edit Site Studio content. Without Oracle HTTP Server (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.

14. (Only if **Authentication Method** is **External Application**) From the **Associated External Application** dropdown list, select the application to use to authenticate users with WebCenter Content Server.

> **Tip:**
>
> If the application has not yet been registered with WebCenter Portal, select **Create New** to register it now. For more information, see Registering External Applications Using Fusion Middleware Control.

15. In the **Administrator User Name** field, enter the user name of a user with administrative rights for this WebCenter Content Server instance. For example, `wccadmin`.

    This user is used to fetch content type information based on profiles and track document changes for cache invalidation purposes.

16. In the **Administrator Password** field, enter the password for the user specified in the **Administrator User Name** field.

17. In the **Cache Invalidation Interval** field, enter the time (in minutes) to allow between checks for external WebCenter Content Server content changes.

    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 is made for content changes (shown as `0`).

18. In the **Maximum Cached Document Size**, enter a maximum cacheable size (in bytes) for WebCenter Content Server binary documents.

    Documents larger than this size are not cached by WebCenter Portal.

    The default is `102400` bytes (100KB).

> **Tip:**
>
> Tune this value based on your machine's memory configuration and the types of binary documents that you expect to cache. Be aware that, unless Coherence is enabled, there is no maximum total size for the cache.
>
> If you are using Coherence, you can additionally specify the total amount of memory to be used for binary caches. For this reason, using Coherence for any type of production environment is strongly recommended, and is a requirement for High Availability (HA) environments.

> **Note:**
>
> Most documents stored in WebCenter 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).

19. Click **Test** to verify if the connection you created works.
20. Click **OK** to save the connection.
21. To start using the connection, you must restart the managed server on which WebCenter Portal is deployed (`WC_Portal` by default).

The registered connection is now available to the Content Manager and Content Presenter task flows, which you can add to pages in WebCenter Portal.

Connecting to Oracle WebCenter Content Server Using Secure Socket-Based Communication

The socket protocol communicates to Oracle WebCenter Content Server over the Intradoc socket port. The socket-based communication can also be configured to run over SSL to provide extra security.

Before you can connect to WebCenter Content Server using secure socket-based communication, you must configure SSL on WebCenter Content Server.

To connect to WebCenter Content Server using secure socket-based communication:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal.
2. From the WebCenter Portal menu, select **Settings** and then **Service Configuration**.
3. On the WebCenter Portal Services Configuration page, from the list of services select **Content Repository**.
4. Click **Add**.
5. In the **Connection Name** field, enter a unique name for the WebCenter Content Server connection.

   > **Tip:**
   >
   > The name must be unique (across all connection types) within WebCenter Portal.

6. The **Repository Type** is **Oracle Content Server**.
7. Select **Active Connection** to make this the default WebCenter Content Server connection for WebCenter Portal.

   You can create connections to multiple WebCenter Content Server instances; all connections are used. One connection must be the default connection. The default

connection is the one used by WebCenter Portal to store portal-related documents.

If this is the default WebCenter Content Server connection for WebCenter Portal, some additional configuration is required. For more information, see Setting Connection Properties for the Default Oracle WebCenter Content Server Connection Using Fusion Middleware Control.

> **Note:**
>
> Deselecting this option does not disable the connection. If a connection is no longer required, you must delete the connection.

8. From the **RIDC Socket Type** dropdown list, select **Socket SSL**.
9. In the **Server Host** field, enter the host name of the machine where WebCenter Content Server is running.

   For example: `mycontentserver.example.com`.
10. In the **Server Port** field, enter the port specified for the WebCenter Content Server's `sslincoming` provider.

    This property corresponds to the IntradocServerPort setting in the WebCenter Content Server configuration file, which defaults to `4444`.

> **Tip:**
>
> You can find the current value by logging into WebCenter Content Server and navigating to **Administration > Admin Server > General Configuration > Additional Configuration Variables > IntradocServerPort**.

11. In the **Connection Timeout** field, enter the length of time (in milliseconds) to attempt to log in to WebCenter Content Server before issuing a connection timeout message.

    This value is also used as the socket timeout for the underlying RIDC connection for all service requests.

    If the **Connection Timeout** is not set, the following values are used:

    - **Login timeout** — the default concurrency timeout configured for the `oracle.webcenter.content` resource (30 seconds or 30000 milliseconds).
    - **RIDC socket timeout** — the default RIDC socket timeout (60 seconds or 60000 milliseconds).

> **Tip:**
>
> It is recommended that you do not specify a value less than 60000 milliseconds for the **Connection Timeout**, 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.

12. From the **Authentication Method** dropdown list, select:

- **Identity Propagation** if WebCenter Content Server uses the same identity store as WebCenter Portal to authenticate users.

  If you select this option, you must also specify the appropriate **Web Server context root for Content Server**.

- **External Application** if WebCenter Content Server uses an external application to authenticate users. Select this option if you want to use public, shared, or mapped credentials.

  If you select this option, you must also specify the appropriate **Associated External Application**.

13. (Only if **Authentication Method** is **Identity Propagation**) In the **Web Server context root for Content Server** field, enter the web server context root for WebCenter Content Server if WebCenter Content Server is front-ended with Oracle HTTP Server (OHS).

    Use the format `/contextRoot`. For example, `/cs`.

    Oracle recommends that you access WebCenter Portal through Oracle HTTP Server (OHS) if you want to use Content Presenter to create or edit Site Studio content. Without Oracle HTTP Server (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.

14. (Only if **Authentication Method** is **External Application**) From the **Associated External Application** dropdown list, select the application to use to authenticate users with WebCenter Content Server.

    > **Tip:**
    >
    > If the application has not yet been registered with WebCenter Portal, select **Create New** to register it now. For more information, see Registering External Applications Using Fusion Middleware Control.

15. In the **Administrator User Name** field, enter a user name with administrative rights for this WebCenter Content Server instance. For example, `wccadmin`.

    This user is used to fetch content type information based on profiles and track document changes for cache invalidation purposes.

16. In the **Administrator Password** field, enter the password for the user specified in the **Administrator User Name** field.

17. In the **Key Store Location** field, enter the location of the keystore that contains the private key used to sign the security assertions.

    The keystore location must be an absolute path.

    For example, `D:\keys\keystore.xyz`.

18. In the **Key Store Password** field, enter the password required to access the keystore.

    For example, `T0PS3CR3T`.

19. In the **Private Key Alias** field, enter the client private key alias in the keystore.

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`.

20. In the **Private Key Password** field, enter the password to use with the private key alias in the keystore.

    For example, `c0d3bR3ak3R`.

21. In the **Cache Invalidation Interval** field, enter the time (in minutes) to allow between checks for external WebCenter Content Server content changes.

    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 is made for content changes (shown as `0`).

22. In the **Maximum Cached Document Size**, enter a maximum cacheable size (in bytes) for WebCenter Content Server binary documents.

    Documents larger than this size are not cached by WebCenter Portal.

    The default is `102400` bytes (100KB).

> **Tip:**
>
> Tune this value based on your machine's memory configuration and the types of binary documents that you expect to cache. Be aware that, unless Coherence is enabled, there is no maximum total size for the cache.
>
> If you are using Coherence, you can additionally specify the total amount of memory to be used for binary caches. For this reason, using Coherence for any type of production environment is strongly recommended, and is a requirement for High Availability (HA) environments.

> **Note:**
>
> Most documents stored in WebCenter 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).

23. Click **Test** to verify if the connection you created works.
24. Click **OK** to save the connection.
25. To start using the connection, restart the managed server on which WebCenter Portal is deployed (`WC_Portal` by default).

The registered connection is now available to the Content Manager and Content Presenter task flows, which you can add to pages in WebCenter Portal.

Connecting to Oracle WebCenter Content Server Using JAX-WS

With JAX-WS, Oracle WebCenter Portal must authenticate with Oracle WebCenter Content Server for each connection rather than assuming any connection from WebCenter Portal is automatically a trusted connection.

To connect to WebCenter Content Server using JAX-WS:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal.
2. From the WebCenter Portal menu, select **Settings** and then **Service Configuration**.
3. On the WebCenter Portal Services Configuration page, from the list of services select **Content Repository**.
4. Click **Add**.
5. In the **Connection Name** field, enter a unique name for the WebCenter Content Server connection.

   > **Tip:**
   >
   > The name must be unique (across all connection types) within WebCenter Portal.

6. The **Repository Type** is **Oracle Content Server**.
7. Select **Active Connection** to make this the default WebCenter Content Server connection for WebCenter Portal.

   You can create connections to multiple WebCenter Content Server instances; all connections are used. One connection must be the default connection. The default connection is the one used by WebCenter Portal to store portal-related documents.

   If this is the default content repository for WebCenter Portal, some additional configuration is required. For more information, see Setting Connection Properties for the Default Oracle WebCenter Content Server Connection Using Fusion Middleware Control.

   > **Note:**
   >
   > Deselecting this option does not disable the connection. If a connection is no longer required, you must delete the connection.

8. From the **RIDC Socket Type** dropdown list, select **JAX-WS**.
9. In the **Web Service URL** field, enter the web service URL required to connect to WebCenter Content Server when using the JAX-WS protocol.

   Use the format `http://host:port/webRoot`

   For example, `http://myhost.com:9044/idcnativews`

10. In the **Client Security Policy** field, enter the client security policy to use.

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 WebCenter Content Server's Native Web Services IdcWebLogin service.

> **Tip:**
>
> Leave this field blank if your environment supports Global Policy Attachments (GPA).

11. In the **Connection Timeout** field, specify the length of time (in milliseconds) to attempt to log in to WebCenter Content Server before issuing a connection timeout message.

    If the Connection Timeout is not set, the default concurrency timeout configured for the `oracle.webcenter.content` resource is used (30 seconds or 30000 milliseconds).

12. In the **Administrator User Name** field, enter a user name with administrative rights for this WebCenter Content Server instance. For example, `wccadmin`.

    This user is used to fetch content type information based on profiles and track document changes for cache invalidation purposes.

13. In the **Administrator Password** field, enter the password for the user specified in the **Administrator User Name** field.

14. In the **Cache Invalidation Interval** field, enter the time (in minutes) to allow between checks for external WebCenter Content Server content changes.

    WebCenter Portal automatically clears items that have changed form the cache. The minimum interval is `2` minutes.

    By default, cache invalidation is disabled so no periodic check is made for content changes (shown as `0`).

15. In the **Maximum Cached Document Size**, enter a maximum cacheable size (in bytes) for WebCenter Content Server binary documents.

    Documents larger than this size are not cached by WebCenter Portal.

    The default is `102400` bytes (100KB).

> **Tip:**
>
> Tune this value based on your machine's memory configuration and the types of binary documents that you expect to cache. Be aware that, unless Coherence is enabled, there is no maximum total size for the cache.
>
> If you are using Coherence, you can additionally specify the total amount of memory to be used for binary caches. For this reason, using Coherence for any type of production environment is strongly recommended, and is a requirement for High Availability (HA) environments.

> **Note:**
>
> Most documents stored in WebCenter 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).

16. Click **Test** to verify if the connection you created works.
17. Click **OK** to save the connection.
18. To start using the connection, restart the managed server on which WebCenter Portal is deployed (`WC_Portal` by default).

The registered connection is now available to the Content Manager and Content Presenter task flows, which you can add to pages in WebCenter Portal.

Connecting to Oracle WebCenter Content Server Using HTTP

Using the HTTPClient package, RIDC communicates with the web server attached to Oracle WebCenter Content Server.

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal.
2. From the WebCenter Portal menu, select **Settings** and then **Service Configuration**.
3. On the WebCenter Portal Services Configuration page, from the list of services select **Content Repository**.
4. Click **Add**.
5. In the **Connection Name** field, enter a unique name for the WebCenter Content Server connection.

   > **Tip:**
   >
   > The name must be unique (across all connection types) within WebCenter Portal.

6. The **Repository Type** is **Oracle Content Server**.
7. Do not select **Active Connection**.

   The HTTP protocol does not allow identity propagation, therefore it is not suitable to use for the default WebCenter Content Server connection for WebCenter Portal. That is, you should not use this protocol to connect to the back-end WebCenter Content Server repository that is being used to store portal-related documents.
8. From the **RIDC Socket Type** dropdown list, select **Web**.
9. In the **Web URL** field, enter the web server URL for WebCenter Content Server.

   Use the format `http://host:port/webRoot/pluginRoot`.

   For example, `http://mycontentserver/cms/idcplug`

.

10. In the **Connection Timeout** field, specify the length of time (in milliseconds) to attempt to log in to WebCenter Content Server before issuing a connection timeout message.

    This value is also used as the socket timeout for the underlying RIDC connection for all service requests.

    If the **Connection Timeout** is not set, the following values are used:

    - **Login timeout** — the default concurrency timeout configured for the `oracle.webcenter.content` resource (30 seconds or 30000 milliseconds).
    - **RIDC socket timeout** — the default RIDC socket timeout (60 seconds or 60000 milliseconds).

    > **Tip:**
    >
    > It is recommended that you do not specify a value less than 60000 milliseconds for the **Connection Timeout**, 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.

11. From the **Associated External Application** dropdown list, select the application to use to authenticate users with WebCenter Content Server.

    > **Tip:**
    >
    > If the application has not yet been registered with WebCenter Portal, select **Create New** to register it now. For more information, see Registering External Applications Using Fusion Middleware Control.

12. In the **Administrator User Name** field, enter a user name with administrative rights for this WebCenter Content Server instance. For example, `wccadmin`.

    This user is used to fetch content type information based on profiles and track document changes for cache invalidation purposes.

13. In the **Administrator Password** field, enter the password for the user specified in the **Administrator User Name** field.

14. In the **Cache Invalidation Interval** field, enter the time (in minutes) to allow between checks for external WebCenter Content Server content changes.

    WebCenter Portal automatically clears items that have changed form the cache. The minimum interval is `2` minutes.

    By default, cache invalidation is disabled so no periodic check is made for content changes (shown as `0`).

15. In the **Maximum Cached Document Size**, enter a maximum cacheable size (in bytes) for WebCenter Content Server binary documents.

    Documents larger than this size are not cached by WebCenter Portal.

    The default is `102400` bytes (100KB).

> **Tip:**
>
> Tune this value based on your machine's memory configuration and the types of binary documents that you expect to cache. Be aware that, unless Coherence is enabled, there is no maximum total size for the cache.
>
> If you are using Coherence, you can additionally specify the total amount of memory to be used for binary caches. For this reason, using Coherence for any type of production environment is strongly recommended, and is a requirement for High Availability (HA) environments.

> **Note:**
>
> Most documents stored in WebCenter 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).

16. Click **OK** to save the connection.
17. Click **Test** to verify if the connection you created works.
18. To start using the connection, restart the managed server on which WebCenter Portal is deployed (`WC_Portal` by default).

The registered connection is now available to the Content Manager and Content Presenter task flows, which you can add to pages in WebCenter Portal.

Registering Oracle WebCenter Content Server Using WLST

Use the WLST command `createContentServerConnection` to register Oracle WebCenter Content Server with Oracle WebCenter Portal.

For command syntax and examples, see createContentServerConnection in *WebCenter WLST Command Reference Reference*.

To configure the connection as the default connection for WebCenter Portal, set `isPrimary='true'`. If you mark a connection as primary, you must run the `setContentServerProperties` WLST command to specify certain additional properties required for the primary WebCenter Content Server connection. See Setting Connection Properties for the Default Oracle WebCenter Content Server Connection Using WLST.

To start using the new connection, you must restart the managed server on which WebCenter Portal is deployed.

Note that if you are using the Content Manager or Content Presenter task flows, WebCenter Content Server should be started first to allow for initial provisioning to take place.

Oracle WebCenter Content Server Connection Parameters for RIDC Socket Types

The configuration parameters you need to set for your Oracle WebCenter Content Server connection depend on the RIDC socket type.

Table 6-4 WebCenter Content Server Connection Parameters for RIDC Socket Types

Connection Parameter (WLST Command Argument)	RIDC Socket Type: HTTP	RIDC Socket Type: Socket	RIDC Socket Type: Secure Socket	RIDC Socket Type: JAX-WS
Connection Name (`name`)	Mandatory	Mandatory	Mandatory	Mandatory
Repository Type	Oracle Content Server	Oracle Content Server	Oracle Content Server	Oracle Content Server
Active Connection (`isPrimary`)	Optional	Optional	Optional	Optional
RIDC Socket Type (`socketType`)	Web	Socket	Socket SSL	JAX-WS
Server Host (`serverHost`)	Not Applicable	Mandatory	Mandatory	Not Applicable
Server Port (`serverPort`)	Not Applicable	Mandatory Defaults to `4444`	Mandatory Defaults to `4444`	Not Applicable
Web URL (`url`)	Mandatory Use the format `http://`*`host`*`:`*`port`*`/`*`webRoot`*`/`*`pluginRoot`*	Not Applicable	Not Applicable	Not Applicable
Web Service URL (`url`)	Not Applicable	Not Applicable	Not Applicable	Mandatory Use the format `http://`*`host`*`:`*`port`*`/`*`webRoot`*

Table 6-4 (Cont.) WebCenter Content Server Connection Parameters for RIDC Socket Types

Connection Parameter (WLST Command Argument)	RIDC Socket Type: HTTP	RIDC Socket Type: Socket	RIDC Socket Type: Secure Socket	RIDC Socket Type: JAX-WS
Client Security Policy (`clientSecurityPolicy`)	Not Applicable	Not Applicable	Not Applicable	Mandatory, unless Global Policy Attachment (GPA) is used, in which case it should be left empty Must match the corresponding server side policy configured for the Content Server's Native Web Services IdcWebLogin service
Connection Timeout (`timeout`)	Optional Do not specify a value less than `60000` (ms) Defaults to `30000` (ms) for login timeout and `60000` (ms) for RIDC socket timeout	Optional Do not specify a value less than `60000` (ms) Defaults to `30000` (ms) for login timeout and `60000` (ms) for RIDC socket timeout	Optional Do not specify a value less than `60000` (ms) Defaults to `30000` (ms) for login timeout and `60000` (ms) for RIDC socket timeout	Optional Defaults to `30000` (ms)
Authentication Method	Not Applicable	Mandatory	Mandatory	Not Applicable
Web Server Context Root (`webContextRoot`)	Not Applicable	Mandatory if Authentication Method is set to Identity Propagation Not Applicable if Authentication Method is set to External Application	Mandatory if Authentication Method is set to Identity Propagation Not Applicable if Authentication Method is set to External Application	Not Applicable
Associated External Application (`extAppId`)	Mandatory	Mandatory if Authentication Method set to External Application Not Applicable if Authentication Method set to Identity Propagation	Mandatory if Authentication Method set to External Application Not Applicable if Authentication Method set to Identity Propagation	Not Applicable

Table 6-4 (Cont.) WebCenter Content Server Connection Parameters for RIDC Socket Types

Connection Parameter (WLST Command Argument)	RIDC Socket Type: HTTP	RIDC Socket Type: Socket	RIDC Socket Type: Secure Socket	RIDC Socket Type: JAX-WS
Administrator User Name (`adminUserName`)	Mandatory `wccadmin`	Mandatory `wccadmin`	Mandatory `wccadmin`	Mandatory `wccadmin`
Administrator Password (`adminPassword`)	Mandatory	Not Applicable	Not Applicable	Optional Whether the password is used or not depends on the selected JAX-WS security policy
Key Store Location (`keystoreLocation`)	Not Applicable	Not Applicable	Mandatory	Not Applicable
Key Store Password (`keystorePassword`)	Not Applicable	Not Applicable	Mandatory	Not Applicable
Private Key Alias (`privateKeyAlias`)	Not Applicable	Not Applicable	Mandatory	Not Applicable
Private Key Password (`privateKeyPassword`)	Not Applicable	Not Applicable	Mandatory	Not Applicable
Cache Invalidation Interval (`cacheInvalidationInterval`)	Optional Defaults to `0` (disabled)	Optional Defaults to `0` (disabled)	Optional Defaults to `0` (disabled)	Optional Defaults to `0` (disabled)
Maximum Cached Document Size (`binaryCacheMaxEntrySize`)	Optional Defaults to `102400` bytes	Optional Defaults to `102400` bytes	Optional Defaults to `102400` bytes	Optional Defaults to `102400` bytes

Configuring Oracle WebCenter Content Server

After installing or upgrading to Oracle WebCenter Content 12*c*, there are several configuration tasks to perform to ensure that Oracle WebCenter Content Server works with Oracle WebCenter Portal.

The configuration tasks are listed in Table 6-3.

> **Note:**
>
> Prior to beginning the configuration you must have completed the installation and configuration steps described in Prerequisites for Configuring Oracle WebCenter Content Server, which define the starting point for the configuration steps in this section.

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

This section includes the following topics:

- Creating a Custom Administrator
- Enabling Mandatory Components
- Configuring the Inbound Refinery
- Setting Up SSL for Oracle WebCenter Content Server
- Setting Up Site Studio
- Enabling Full-Text Search
- Creating Content Profiles in Oracle WebCenter Content Server
- Enabling Digital Asset Manager
- Additional Optional Configurations for Oracle WebCenter Content Server
- Registering the Default Oracle WebCenter Content Server Repository

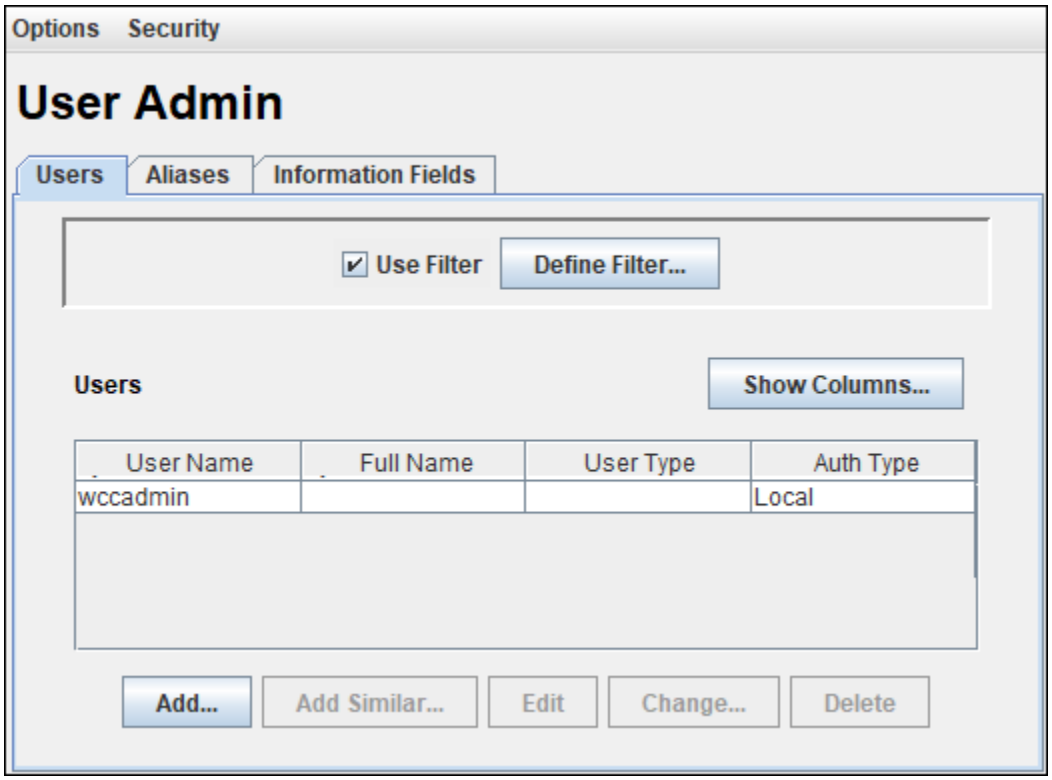
Creating a Custom Administrator

To create the Content Server connection, you will have to use a custom administrator.

Before creating the connection, you should create a custom administrator using the following steps:

1. Log in to the Content Server UI as an administrator.
2. On the WebCenter Content home page, select **Admin Applets** from the Administration menu.
3. Select **User Admin**.
   The User Admin dialog opens.

Figure 6-1 User Admin dialog



4. On the Users tab, click **Add**.
5. Choose the Authentication Type as **Local** and click **OK**.
6. In the Add User dialog, do the following:
   a. On the Info tab, enter the name as `wccadmin`, specify password and confirm password.
   b. On the Roles tab, click **Add Role** and select **admin** and **sysmanager** to assign the role to the user.
   c. On the Accounts tab, remove the row for **[documents without accounts]**. Select the row and then click **Delete**.
   Click **Add**. In the Add New Account dialog, select **Account** as **[all accounts]** and then click **OK**.

Enabling Mandatory Components

A component is a functional unit that can be plugged into Oracle WebCenter Content Server to provide additional features or to modify existing functionality.

To prepare WebCenter Content Server for Oracle WebCenter Portal, you must:

- Enable FrameworkFolders

  For information, see Enabling the FrameworkFolders Component
- Enable WebCenterConfigure

  For information, see Enabling the WebCenterConfigure Component
- Enable DynamicConverter

For enabling the component, follow the same procedure that you used for enabling other components, such as FrameworkFolders and WebCenterConfigure.

Enabling the FrameworkFolders Component

FrameworkFolders provides a hierarchical folder interface similar to a conventional file system, for organizing and locating some or all of the content in Oracle WebCenter Content Server. In addition, it enables you to use WebCenter Content Server mobile applications to access content in portals and leverage the ADF content UI.

To enable the FrameworkFolders component:

1. Log on to WebCenter Content Server as an administrator.
2. From the **Main** menu, choose **Administration**, then **Admin Server**, then **Component Manager**.
3. On the Component Manager page, select the **FrameworkFolders** check box.
4. Click **Update**.
5. Click **Advanced Component Manager**.
6. On the Advanced Component Manager page, ensure that:
   - **FrameworkFolders** is listed in the Enabled Components section
   - **Folders_g** is listed in the Disabled Components section
7. Restart the WebCenter Content Server instance.

Enabling the WebCenterConfigure Component

You must enable the WebCenterConfigure component to configure Oracle WebCenter Content Server for Oracle WebCenter Portal.

Table 6-5 describes the tasks performed in WebCenter Content Server when you enable this component.

To enable the WebCenterConfigure component:

1. Log on to WebCenter Content Server as an administrator.
2. From the **Main** menu, choose **Administration**, then **Admin Server**, then **Component Manager**.
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 **Digital Asset** if you plan to use them as you'll otherwise need to enable them later.

4. Click **Update**.
5. Click **Advanced Component Manager**.
6. On the Advanced Component Manager page, ensure that WebCenter Configure is listed in the Enabled Components section.

7. Restart the WebCenter Content Server instance.

Enabling the WebCenterConfigure component performs the following tasks in WebCenter Content Server:

Table 6-5 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`
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.
Adds the `WCWorkflowApproverUserToken` workflow token	Content Server > Administration > Admin Applets > Workflow Admin > Options > Tokens menu
Overrides certain behavior of the Site Studio Switch Content wizard to make Site Studio work in WebCenter Portal	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. • 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.

Configuring the Inbound Refinery

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.

Optional, but strongly recommended

You can use Inbound Refinery to convert content items stored in Oracle WebCenter 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 WebCenter Content Server to Inbound Refinery, and specify the file types that will be converted. 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, and completed the initial post-install configuration as described in Configuration Prerequisites for Oracle WebCenter Content Server and Inbound Refinery.

This section contains the following subsections:

- Creating an Outbound Provider
- Selecting the File Formats To Be Converted
- Enabling the Conversion of Wikis and Blogs into PDFs
- Specifying the Timeout Setting for File Conversions

Creating an Outbound Provider

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

To create an outbound provider:

1. From the WebCenter 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:

  a. Log in to WebCenter Content Server and choose **Administration > Configuration for *instanceName***.

  b. 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 WebCenter 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 WebCenter Content Server and Inbound Refinery instances can ping each other.

Selecting the File Formats To Be Converted

To tell Oracle WebCenter 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 WebCenter 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.

   WebCenter 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 WebCenter Content Server.

2. Select the file formats that you want to be converted.

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

3. Click **Update**.

Enabling the Conversion of Wikis and Blogs into PDFs

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 Oracle WebCenter Content Server respectively.

Optional

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 Using the Rich Text Editor (RTE) in *Using Portals in Oracle WebCenter Portal*.
>
> See also, File Formats Converted to PDF by Open Office in *Managing Oracle WebCenter Content*.

Before you can enable conversion of wikis and blogs into PDFs in WebCenter Portal, ensure you have done the following:

- Set up the OpenOffice integration with Inbound Refinery. See Configuring Inbound Refinery to Use OpenOffice in *Managing Oracle WebCenter Content*.
- Set up the path to the OpenOffice class files. See Setting Classpath to OpenOffice Class Files in *Managing Oracle WebCenter Content*.

To enable conversion of wikis and blogs into PDFs in WebCenter Portal:

1. Install the WebCenterConversion component:
   a. Log in to the Inbound Refinery server.
   b. Click **Administration** and then select **Admin Server**.

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

      The Advanced Component Manager page displays.

d. In the Install New Component section, select the path to the `WebCenterConversions.zip`, then click **Install**.

   The `WebCenterConversions.zip` can be found under *`MW_HOME`*`/wcportal/install/`.

   The WebCenterConversion component displays in the Disabled Components box.

e. Select **WebCenterConversion** and click **Enable**.

f. Restart the Inbound Refinery server.

2. Enable Inbound Refinery to integrate with OpenOffice:

   a. Log in to the Inbound Refinery server again.

   b. Click **Administration** and then select **Admin Server**.

      The Inbound Refinery Admin Server page displays.

   c. In the Component Manager, select the **OpenOfficeConversion** check box.

   d. Restart the Inbound Refinery server.

3. Enable the WebCenterConversion component:

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

      This displays the Conversion Listing page.

   b. In the **Conversions** table, select the **Accept** check box for `HtmToPDFOpenOffice`, and click **Update**.

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

4. Enable Wikis and Blogs to be converted to PDFs in WebCenter Content Server:

   a. Log in to WebCenter Content Server.

   b. Expand the **Administration** node, then **Refinery Administration**, and then click **File Formats Wizard**.

   c. Under **Select File Types**, select the **Wiki** and **Blogs** check boxes and click **Update**.

5. Enable the PDF conversion in Inbound Refinery:

   a. Log in to the Inbound Refinery server again.

   b. Select **Conversion Settings**, and then select **Primary Web Rendition**.

   c. Check the **Convert to PDF using Open Office** option.

   d. Click **Update**.

Specifying the Timeout Setting for File Conversions

You can optionally set the maximum and minimum amount of time for which Inbound Refinery will process the different conversion operations.

To set the timeout settings for conversion operations:

1. Log on to the Inbound Refinery server.
2. Under **Settings**, select **Timeouts**.
3. Specify the minimum and maximum timeout settings for the various conversion operations as required.

4. Click **Update**.

> **Note:**
>
> For information about setting the timeout settings in Dynamic Converter, see #unique_205.

Setting Up SSL for Oracle WebCenter Content Server

If Oracle WebCenter Portal and the Oracle WebCenter Content Server you intend to use for your content repository 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 WebCenter Content Server.

Setting Up Site Studio

Configuring Site Studio lets you use Site Studio to create and use Site Studio assets (region definitions and display templates) in Content Presenter.

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 WebCenter 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)
   - PortalVCRHelper

4. Click **Update**.
5. Restart the WebCenter Content Server instance.
6. Log back into WebCenter 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. To use the Site Studio Designer, log into the WebCenter Content Server console, navigate to **My Content Server > My Downloads**, then download and install Site Studio Designer.

After setting up Site Studio, start (or restart) Oracle WebCenter Portal to seed the WebCenter Content Server instance with the appropriate assets, such as the `RD_ARTICLE` region definition.

Enabling the iFraming UI

If you want Site Studio to be displayed in Content Presenter using inline frames rather than in separate windows, and Oracle WebCenter Portal and Oracle WebCenter Content Server are not in the same domain (in terms of their web address), you must configure the Oracle HTTP Server (OHS).

> **Notes:**
>
> - Before enabling support for iFraming, you should already have installed and configured OHS as described in Installing and Configuring Oracle HTTP Server.
> - While Content Presenter allows specifying a different Content Server connection, iFraming is supported only for the default Content Server connection.

To enable the iFraming UI:

1. Open the `mod_wl_ohs.conf` file and make sure it points to the right WebCenter 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:**
   >
   > This setting should never be set if OHS is not set up or is not working correctly.
3. Configure OHS by updating the `mod_wl_ohs.conf` file with the WebCenter 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 WebCenter 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>
```

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

`http://host:OHSPort/webcenter`

For example:

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

Enabling Full-Text Search

By default, the database used by Oracle WebCenter 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.

Optional, but strongly recommended

Although nominally optional, Oracle recommends that you implement full-text search using the OracleTextSearch option.

For more information, see Configuring OracleTextSearch for Content Server in *Installing and Configuring Oracle WebCenter Content*, and Site Studio Integration in *Managing Oracle WebCenter Content*.

Creating Content Profiles in Oracle WebCenter Content Server

Users have the option to upload content using Content Server Profiles.

Optional

For more information on WebCenter Content Server Profiles, see Managing Content Profiles in *Managing Oracle WebCenter Content*.

You can use the content check-in page to check files into WebCenter Content Server. Required fields are indicated by an asterisk (*). All content profiles must include the mandatory fields, otherwise the check-in will fail.

In addition to the mandatory fields needed to upload files to WebCenter Content Server, for the upload profiles to work correctly in Document Library and Oracle WebCenter Portal, the WebCenter 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

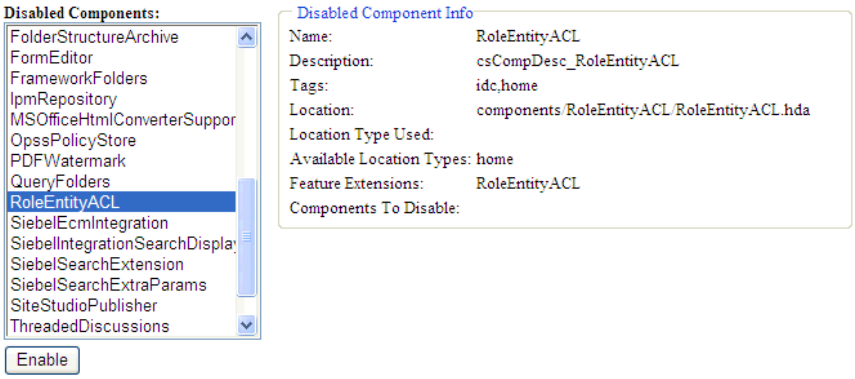
These fields can be added as hidden fields to the profile.

Configuring Access Control List Security

To configure access control lists (ACLs):

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**, and then click **Enable**.

Figure 6-2 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. ACLs are enabled only on the content in the specified security groups.

The name of the security group that contains the data is the same as application name you configured to identify 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.

7. Restart Content Server and the managed server on which WebCenter Portal is running.

Enabling Digital Asset Manager

For full image rendition support, the Oracle WebCenter Content Server where your images are checked in must have Digital Asset Manager (DAM) enabled.

Optional

For example, you may want to use a large, high resolution image when the page containing the image is displayed using a desktop browser; a smaller, lower resolution image for display on a mobile phone; and a medium-sized, but still low resolution image for display on a tablet.

When DAM is enabled, different renditions are automatically created when an image is checked in, determined by the rendition set specified during check in. DAM provides some built-in rendition sets but the Content Server administrator can also create new rendition sets. The individual renditions can then be referenced by name in Content Presenter display templates by using the appropriate EL expression.

If DAM is not enabled, there is limited support only for image renditions through Inbound Refinery with `web` and `thumbnail` renditions.

For more information about enabling DAM and creating rendition sets, see Working with Image and Video Conversions in *Managing Oracle WebCenter Content*.

> **Note:**
>
> Oracle WebCenter Portal supports multiple renditions for images only, not video.

Additional Optional Configurations for Oracle WebCenter Content Server

This section describes additional optional configurations that are not required for Oracle WebCenter Content Server to function correctly, but nonetheless offer value and comprise best practices for a WebCenter Content Server enterprise installation.

This section includes the following topics:

- Configuring Oracle WebCenter Content Server for Desktop
- Configuring the File Store Provider
- Setting Up Node Manager
- Configuring Localization Properties

- Showing and Hiding the Wiki Markup Tab in the Rich Text Editor
- Disabling Text Wrapping in the Rich Text Editor

Configuring Oracle WebCenter Content Server for Desktop

Oracle WebCenter Content: Desktop provides convenient access to Oracle WebCenter Content Server files from a number of familiar desktop applications, such as Windows Explorer, Microsoft Office applications (Word, Excel, and Powerpoint), email clients (Microsoft Outlook and Lotus Notes), and web browsers (Internet Explorer, Mozilla Firefox, and Google Chrome).

For the Desktop client software to connect to WebCenter Content Server, the following system component must be enabled on the server:

- `CoreWebdav`, which provides core WebDAV capabilities for the content management integrations.

In addition, you must also enable the following components:

- `DesktopIntegrationSuite`, which handles core content management integration functions on the server.
- `DesktopTag`, which manages custom properties in Microsoft Office files that are used for content tracking purposes, and also provides the workflow processing functionality in Microsoft Office applications.
- `Framework Folders`, which enables the content folders in the integration hierarchy.

You can also enable the following component:

- `EmailMetadata`, which maps email message fields to email metadata fields and is also required for dragging and dropping emails into content folders in Microsoft Outlook and Lotus Notes.

To configure WebCenter Content Server for Desktop:

1. Log in to WebCenter Content Server.
2. In the WebCenter Content Server **Administration** tray or menu, choose an **Admin Server**, then **Component Manager**.
3. On the Component Manager page, select **Folders** to display the Folders category of components.
4. Select the **FrameworkFolders** component.
5. Select the **DesktopIntegrationSuite**, **DesktopTag**, and, optionally, the **EmailMetadata** components.
6. Click the **Update** button, and then click **OK** to confirm your selections.
7. In the first paragraph of the Component Manager page, click **advanced component manager**.
8. In the Disabled Components box on the Advanced Component Manager page, select **FolderStructureArchive**, and click the **Enable** button.
9. Make sure that the `CoreWebdav` component is enabled:
    a. Under Category Filters on the Advanced Component Manager page, select **Show System Components**.

b. If **CoreWebdav** is not in the Enabled Components box, select **CoreWebdav** in the Disabled Components box, and click the **Enable** button.

10. Restart WebCenter Content Server.

For additional configuration information for Desktop, see Managing Desktop in *Managing Oracle WebCenter Content*.

The Desktop client software must be installed on the computers of users wishing to use desktop integration. For more information, see Setting Up the Desktop Client Software on Your Computer in *Using Oracle WebCenter Content: Desktop*.

Configuring the File Store Provider

A file store for data management is used in Oracle WebCenter Content Server 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 WebCenter 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 WebCenter Content Server interface and allows additional configuration options. For example, you can configure the WebCenter 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 WebCenter 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 WebCenter 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 WebCenter 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 WebCenter Content Server instance.

Setting Up Node Manager

As an additional step to configuring and managing Oracle WebCenter 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 WebCenter 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 Using Node Manager with Oracle WebCenter Content in *Installing and Configuring Oracle WebCenter Content*.

Configuring Localization Properties

To ensure the Content Manager task flow works and displays proper translations on non-English locales, you need to configure Content Server for localization.

To configure enable or disable locales on Content Server:

1. Log on to WebCenter Content as an administrator.
2. From the Main menu, choose **Administration**, and then **Localization**.
3. Select the check boxes for the required locales.
4. Click **Update**.

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

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`.

Optional

> **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.

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_Portal', toLocation='/scratch/
aime1', docs='/oracle/webcenter/doclib/config/mdssys/cust/site/webcenter/blog-wiki-
config.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|false]"/></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_Portal', fromLocation='/
scratch/aime1', docs='/oracle/webcenter/doclib/config/mdssys/cust/site/
webcenter/blog-wiki-config.xml.xml')
```

Disabling Text Wrapping in the Rich Text Editor

By default, the Rich Text Editor wraps HTML source at 68 characters. This may cause some multibyte symbols to wrap incorrectly. If you encounter this problem, you can disable text wrapping in the Rich Text Editor.

Optional

To disable text wrapping in the Rich Text Editor:

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

```
exportMetadata(application='webcenter', server='WC_Portal', toLocation='/
scratch/aime1', docs='/oracle/webcenter/doclib/config/mdssys/cust/site/
webcenter/blog-wiki-config.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 `text.wrap.length` to `0`:

```
<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='text.wrap.length']"><mds:attribute name="value"
value="0"/></mds:modify>
```

4. Import the updated file to MDS:

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

Registering the Default Oracle WebCenter Content Server Repository

The default connection between Oracle WebCenter Portal and Oracle WebCenter Content Server may be configured for you when WebCenter Portal first starts up, but Oracle strongly recommends that you test the connection and check that the expected data has been properly seeded.

Optional, but strongly recommended

This section includes the following topics:

- Configuring the Default Oracle WebCenter Content Server Connection for Oracle WebCenter Portal
- Checking the Oracle WebCenter Portal Data Seeded in Oracle WebCenter Content Server

Configuring the Default Oracle WebCenter Content Server Connection for Oracle WebCenter Portal

A default connection between Oracle WebCenter Portal and Oracle WebCenter Content Server may be automatically configured when WebCenter Portal 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 WebCenter Content Server, and restarting WebCenter Portal, check the connection between WebCenter Portal and WebCenter Content Server is properly configured. If your connection was not properly configured, then configure it as shown in Setting Connection Properties for the Default Oracle WebCenter Content Server Connection.

Some WebCenter Portal components rely on the data seeded in WebCenter Content Server when WebCenter Portal first starts up. Before configuring other components with WebCenter Portal, check that the expected data has been properly seeded.

Checking the Oracle WebCenter Portal Data Seeded in Oracle WebCenter Content Server

When Oracle WebCenter Portal first starts up, a set of default data is seeded in the default Oracle WebCenter Content Server. The data seeded in WebCenter Content Server for a WebCenter Portal instance is based on several properties that are set on the default WebCenter Content Server connection.

For example:

```
Portal Server Identifier = /WebCenter1
Security Group = WC1
```

If the data is not correct, or has only been partially seeded, check the WebCenter Portal log and your WebCenter 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 WebCenter Content Server connection, and setting additional properties required for WebCenter Portal's content repository, see Setting Connection Properties for the Default Oracle WebCenter Content Server Connection.

Table 6-6 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 WebCenter Content Server (**Verify**).

Table 6-6 Data Seeded in WebCenter Portal

Seeded Data	Naming	Verify
Security Group	One security group is seeded: *securityGroup* For example: WC1	In WebCenter Content Server, go to **Administration > Admin Applets > User Admin > Security> Permission by Group**
Roles	Two roles are seeded: • *securityGroup*User (with R permission on the security group) • *securityGroup*AuthenUser (with RWD permission on the security group) For example: WC1User and WC1AuthenUser	In WebCenter Content Server, go to **Administration > Admin Applets > User Admin > Security > Permission by Role**
Root Folder name	*portalServerIdentifier* (with Security Group = *securityGroup*) 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 *securityGroup* role	Query the ExtendedConfigProperties table, or after logging into WebCenter Content Server, click on the user name to view the user's profile page listing their roles and accounts, including the account PUBLIC and WCILS and the role *securityGroup*User

Table 6-6 (Cont.) Data Seeded in WebCenter Portal

Seeded Data	Naming	Verify
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 *securityGroup*`AuthenUser` role	Query the `ExtendedConfigProperties` table, or after logging into WebCenter Content Server, click on the user name to view the user's profile page listing their roles and accounts, including the account `AUTHEN` and `WCILS` and the role *securityGroup*`AuthenUser`

Table 6-7 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 WebCenter Content Server (**Verify**). Note that the Home portal data is seeded only once in a WebCenter Content Server instance, regardless of how many WebCenter Portal instances are using the same WebCenter Content Server. Therefore, if you have multiple WebCenter Portal instances using the same WebCenter Content Server, they will all share the same Home portal data.

Table 6-7 Data Seeded for the Home Portal

Seeded Data	Naming	Verify
Security Group	One security group is seeded: `PersonalSpaces`	In WebCenter 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 WebCenter 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 WebCenter Content Server, click on the user name to view the user's profile page listing their roles and accounts, including the account `PEWebCenter/PU` and the role `PersonalSpacesRole`

Table 6-7 (Cont.) Data Seeded for the Home Portal

Seeded Data	Naming	Verify
Default Attributes - Authenticated users	All Authenticated users have: • The `PersonalSpacesAuthenRole` role	Query the `ExtendedConfigProperties` table, or after logging into WebCenter Content Server, click on the user name to view the user's profile page listing their roles and accounts, including the role `PersonalSpacesAuthenRole`

Setting Connection Properties for the Default Oracle WebCenter Content Server Connection

The default content repository is the one used by WebCenter Portal to store portal-related documents. Some additional configuration is required for the default repository.

This section contains the following topics:

- Setting Connection Properties for the Default Oracle WebCenter Content Server Connection Using Fusion Middleware Control
- Setting Connection Properties for the Default Oracle WebCenter Content Server Connection Using WLST

Setting Connection Properties for the Default Oracle WebCenter Content Server Connection Using Fusion Middleware Control

You can view, modify, and delete connection information for the Oracle WebCenter Content Server connection that is being used by Oracle WebCenter Portal to store portal documents.

> **WARNING:**
>
> You should never change the **Portal Server Identifier** or **Security Group** values separately; you should always change both. That is, if you change the **Portal Server Identifier** value after configuring and running WebCenter Portal, then you must also change the **Security Group** value, and vice versa. That is, you must change both values (**Portal Server Identifier** and **Security Group**) to unique values if WebCenter Portal already contains the seed data.

To set connection properties for the default WebCenter Content Server connection using Fusion Middleware Control:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal.

2. From the WebCenter Portal menu, select **Settings** and then **Service Configuration**.
3. On the WebCenter Portal Services Configuration page, from the list of services select **Content Repository**.
4. Select the default connection, and click **Edit**.
5. In the **Portal Server Identifier** field, enter a name for the folder in WebCenter Content Server under which WebCenter Portal content is stored.

   Use the format `/folderName`.

   For example `/WebCenter` or `/WCPMain`.

   The folder cannot be the WebCenter Content Server root folder (`/`) and it must be unique across applications. If the folder does not exist it will be created for you.

   > **Note:**
   >
   > When you change this value, the existing seed data is not renamed in WebCenter Content Server. Instead, new seed data is created using the new value when you start WebCenter Portal. Once WebCenter Portal is started, new data is created under the new folder and existing data under the old 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 **Portal Server Identifier**.
   >
   > The old folder still appears in search results, like any other root folder in WebCenter Content Server.

   > **WARNING:**
   >
   > If you change the **Portal Server Identifier** you must also provide a new value for **Security Group**.

6. In the **Content Administrator** field, enter a user name with administrative rights for the WebCenter Content Server instance.

   For example, `wccadmin`.

   This user will be used to create and maintain folders for WebCenter Portal content and manage content access rights. Administrative privileges are required for the default connection so that operations can be performed on behalf of WebCenter Portal users.
7. In the **Security Group** field, enter a unique identifier to use as the value for the security group assigned to files in WebCenter Content Server created in WebCenter Portal.

   This name is used to separate data when multiple WebCenter Portal instances share the same WebCenter Content Server instance.

   The application name must be:

   - Unique across all WebCenter Portal applications.
   - Must begin with an alphabetical character, followed by any combination of alphanumeric characters or the underscore character.
   - Must be less than or equal to 30 characters.

> **WARNING:**
>
> If you change the **Security Group** you must also provide a new value for **Portal Server Identifier**.

8. Click **OK** to save your changes.

To start using the updated connection properties, you must restart the managed server on which WebCenter Portal is deployed (`WC_Portal` by default).

Setting Connection Properties for the Default Oracle WebCenter Content Server Connection Using WLST

Use WLST commands to view, set, and delete properties for the Oracle WebCenter Content Server connection that is being used by Oracle WebCenter Portal to identify where to store portal documents.

The following values must be set for the default WebCenter Content Server connection:

- `portalServerIdentifier`—specify a name for the folder in WebCenter Content Server under which WebCenter Portal content is stored.
- `adminUserName`—specify a user name with administrative rights for the WebCenter Content Server instance.
- `securityGroup`—specify a unique identifier to use as the value for the security group assigned to files in WebCenter Content Server created in WebCenter Portal.

> **WARNING:**
>
> You should never change the `portalServerIdentifier` or `securityGroup` values separately; you should always change both. That is, if you change the `portalServerIdentifier` value after configuring and running WebCenter Portal, then you must also change the `securityGroup` value, and vice versa. That is, you must change both values (`portalServerIdentifier` and `securityGroup`) to unique values if WebCenter Portal already contains the seed data.

Use the following commands (for command syntax and detailed examples, see the linked section in *WebCenter WLST Command Reference Reference*):

- listContentServerProperties
- setContentServerProperties
- deleteContentServerProperties

Modifying Oracle WebCenter Content Server Connection Details

This section contains the following topics:

- Modifying Oracle WebCenter Content Server Connection Details Using Fusion Middleware Control
- Modifying Oracle WebCenter Content Server Connection Details Using WLST
- Modifying Cache Settings for Content Presenter
- Configuring the Cache to Check for External Oracle WebCenter Content Server Changes

Modifying Oracle WebCenter Content Server Connection Details Using Fusion Middleware Control

You can modify Oracle WebCenter Content Server connection details using Fusion Middleware Control.

To update WebCenter Content Server connection details using Fusion Middleware Control:

1. Log in to Fusion Middleware Control and navigate to the home page for Oracle WebCenter Portal.
2. From the WebCenter Portal menu, select **Settings** and then **Service Configuration**.
3. On the WebCenter Portal Services Configuration page, from the list of services select **Content Repository**.
4. Select the connection name, and click **Edit**.
5. Edit connection details, as required.

   For detailed parameter information, see Creating a Connection to Oracle WebCenter Content Server Using Fusion Middleware Control.
6. Click **Test** to verify if the updated connection works.
7. Click **OK** to save your changes.
8. To start using the updated connection, you must restart the managed server on which WebCenter Portal is deployed.

Modifying Oracle WebCenter Content Server Connection Details Using WLST

Use the WLST command `setContentServerConnection` to edit Oracle WebCenter Content Server connection details.

For command syntax and examples, see setContentServerConnection in *WebCenter WLST Command Reference Reference*.

To configure a particular connection as the default connection, set `isPrimary='true'`. See Setting Connection Properties for the Default Oracle WebCenter Content Server Connection Using WLST.

> **Note:**
>
> To start using the updated connection details, you must restart the managed server on which WebCenter Portal is deployed.

Modifying Cache Settings for Content Presenter

Content Presenter, by default, is configured to use a local (in-memory) cache. Using Coherence for any type of production environment, however, is strongly recommended, and is a requirement for High Availability (HA) environments. You can enable content caching with Coherence by modifying the Coherence configuration file.

> **Note:**
>
> Your Coherence license may or may not support multi-node environments depending on the license option you have purchased.

To enable and test Coherence as the caching mechanism:

1. Open the `ORACLE_HOME/wcportal/webcenter/modules/oracle.webcenter.framework/content-app-lib.ear` file and copy the `sample-content-coherence-cache-config.xml` file from `ORACLE_HOME/wcportal/webcenter/modules/oracle.webcenter.framework/content-app-lib.ear`..

   You'll find the `sample-content-coherence-cache-config.xml` file under `/content-app-lib.ear/APP-INF/classes/sample-content-coherence-cache-config.xml`.

2. Copy the `sample-content-coherence-cache-config.xml` file to `MW_HOME/user_projects/applications/<Domain_Name>/custom.webcenter.spaces.fwk/APP-INF/classes/` and rename it as `content-coherence-cache-config.xml`.

3. Modify the Coherence configuration file for your local environment based on the example file (Example 6-1) and entry descriptions in the following table.

Table 6-8 Cache Entries in content-coherence-cache-config.xml

Cache Entry Name	Description
`repo.ucm.nodeUidCache.*`	Stores a list of nodes for a repository based on an ID. The size of this cache entry depends upon the number of nodes in the active repository. This cache expires based on when the node data is refreshed and how many times the data is modified from another application. **Key** - Node UID - String **Value** - An Oracle WebCenter Content Server Node object

Table 6-8 (Cont.) Cache Entries in content-coherence-cache-config.xml

Cache Entry Name	Description
`repo.ucm.nodePathToUidCache.*`	Stores a list of nodes for a repository based on a path. The size of this cache depends upon the number of nodes in the default repository. This cache entry expires based on when the node data is refreshed and how many times the data is modified from another application. The size and expiration time must be the same as that of `nodeUidCache`. **Key** - Node path - String **Value** - Node UID - String
`repo.ucm.securityInfoCache.*`	Stores cached security information for a node. The size of this cache depends upon the number of nodes in the repository. This cache expires based on the frequency of node security data updates. **Key** - Node UID - String **Value** - Security information for a node
`repo.ucm.typeNameCache.*`	Caches Content Type information. The size of this cache depends upon the number of types in the repository. This cache expires based on when the type information is refreshed and how many times the types are modified from another application. **Key** - Content Type UID - String **Value** - A ContentType object
`repo.ucm.typeNamesCache.*`	Caches all the type names known to WebCenter Content Server. All type names are cached together (one key), and thus all expire at the same time. This cache expires based on the frequency of new types being created or removed. **Key** - There is only one key to this cache: `typeNames` **Value** - An `ArrayList<String>` of the type names

Table 6-8 (Cont.) Cache Entries in content-coherence-cache-config.xml

Cache Entry Name	Description
`binaryCache.*`	Caches binary property data. Only binaries that are smaller than the repository configuration property `binaryCacheMaxEntrySize` are cached. The size of this cache either depends on the number and frequency of the smaller binary properties (smaller than the `binaryCacheMaxEntrySize` setting) usage, or it is based on the total amount of memory to be used for binary caches. This cache expires based on when the binary data is refreshed and how many times this data is modified from another application. **Key** - The Node UID and binary Property UID (`nodeUid.propUid`) - String **Value** - The binary stream data - `byte[]` **Note:** Most documents stored in WebCenter 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).
`repo.ucm.searchCriteriaCache.*`	Caches a set of search query to parameters based on the WebCenter Content Server search grammar. The size of this cache depends upon the number of unique searches expected to be repeatedly performed. The expiration must be set to eventually expire unused searches and save on the cache memory. **Key** - A set of search query parameters. **Value** - A set of search query parameters, in Content Server terms.

Table 6-8 (Cont.) Cache Entries in content-coherence-cache-config.xml

Cache Entry Name	Description
`repo.ucm.indexedFieldsCache.*`	Holds the indexed (searchable) system properties for the repository. There are three keys in this cache: • `indexedFields` holds all WebCenter Content Server indexed fields. • `indexedFolderProps` holds indexed system properties for folders. • `indexedDocProps` holds indexed system properties for documents. This cache expires based on the frequency of the indexed fields changes. **Key** - String **Value** - `Map<String,Boolean>` holds a key for each indexed property name, and a Boolean indicating if that property is also sortable.
`repo.ucm.securityUserCache.*`	Caches the mapping between local user names (current application) and the name of the same user in WebCenter Content Server. The size of this cache depends upon the number of simultaneous and/or frequent users. This cache expires based on the frequency of user identity mapping updates. **Key** - Local user Id - String **Value** - WebCenter Content Server user Id - String
`repo.ucm.profileTriggerValueCache.*`	Caches the profile trigger value for a given profile, so it is available when documents are created. The maximum number of entries in this cache is implicitly limited to the maximum number of profiles on the WebCenter Content Server instance. The cache entry size is small. The primary entry to vary is the expiration, which depends upon how often the profile trigger field values are modified in WebCenter Content Server. These values change rarely once a profile is configured on the WebCenter Content Server system. Therefore, the expiration should be set appropriately. **Key** - The WebCenter Content Server profile name - String **Value** - The WebCenter Content Server profile trigger value - String

Table 6-8 (Cont.) Cache Entries in content-coherence-cache-config.xml

Cache Entry Name	Description
`repo.ucm.resultOfAQueryCache.*`	Include this parameter when you upgrade WebCenter Portal from 11g to 12c. Caches the result of a given query. The result of a query execution on Content Server depends on the security permission for a given user. The cache is maintained per user, so results are different for different users. Modify the high units and expiry delay parameter as per your requirement. For example: `<expiry-delay>10m</expiry-delay> <high-units>1000</high-units> .` **Key** - the combination of the use and the search query **Value** - the list of WebCenter Content node object
`repo.ucm.contentsUnderAFolderCache.*`	Include this parameter when you upgrade WebCenter Portal from 11g to 12c. Caches the contents under a folder. The content shown to the user depends on the security permission on Content Server. The cache is maintained per user, so the list of content is different for different users. Modify the high units and expiry delay parameter as per your requirement. For example: `<expiry-delay>10m</expiry-delay> <high-units>1000</high-units> .` **Key** - the combination of the use and the folder identifier. **Value** - Array of node object

4. Add the following to the `setDomainEnv.sh` file so that you can test that Coherence has been properly configured:

```
JAVA_OPTIONS="${JAVA_OPTIONS} -Dtangosol.coherence.management=all"
export JAVA_OPTIONS
```

5. Restart the `WC_Portal` server and connect to it by entering `jconsole` from the command line and choosing the process corresponding to `WC_Portal` to open JConsole.
6. In JConsole, check for Coherence in the MBeans tab.

> **Note:**
>
> - There must be something in the cache for the MBeans to appear in Jconsole. That is, you must have created and accessed a Content Presenter page for the MBeans to exist.
> - Once a Content Presenter page exists, thus populating the cache, in JConsole connected to the `WC_Portal` server, you can open **Coherence > Cache > LocalCache** and see multiple entries for `repo.ucm.*.{`*`ucm-connection-name`*`}`. For example, `repo.ucm.typeNameCache.{`*`ucm-connection-name`*`}`

Example 6-1 Sample Coherence Configuration File

```
<!DOCTYPE cache-config SYSTEM "cache-config.dtd">
<cache-config>
  <caching-scheme-mapping>
    <cache-mapping>
      <cache-name>repo.ucm.nodeUidCache.*</cache-name>
      <scheme-name>ContentNodeCaches</scheme-name>
    </cache-mapping>
    <cache-mapping>
      <cache-name>repo.ucm.nodePathToUidCache.*</cache-name>
      <scheme-name>ContentNodeCaches</scheme-name>
    </cache-mapping>
    <cache-mapping>
      <cache-name>repo.ucm.securityInfoCache.*</cache-name>
      <scheme-name>ContentNodeCaches</scheme-name>
    </cache-mapping>
    <cache-mapping>
      <cache-name>repo.ucm.typeNameCache.*</cache-name>
      <scheme-name>ContentTypeCaches</scheme-name>
    </cache-mapping>
   <cache-mapping>
      <cache-name>repo.ucm.typeNamesCache.*</cache-name>
      <scheme-name>ContentTypeCaches</scheme-name>
    </cache-mapping>
    <cache-mapping>
      <cache-name>binaryCache.*</cache-name>
      <scheme-name>ContentBinaryCaches</scheme-name>
    </cache-mapping>
    <cache-mapping>
      <cache-name>repo.ucm.searchCriteriaCache.*</cache-name>
      <scheme-name>ContentSearchCaches</scheme-name>
    </cache-mapping>
    <cache-mapping>
      <cache-name> repo.ucm.indexedFieldsCache.*</cache-name>
      <scheme-name>ContentSearchCaches</scheme-name>
    </cache-mapping>
    <cache-mapping>
      <cache-name>repo.ucm.securityUserCache.*</cache-name>
      <scheme-name>ContentSecurityCaches</scheme-name>
    </cache-mapping>
```

```
    <cache-mapping>
      <cache-name>repo.ucm.profileTriggerValueCache.*</cache-name>
      <scheme-name>ContentProfileCaches</scheme-name>
    </cache-mapping>
    <cache-mapping>
      <cache-name>binaryContentTypeCache.*</cache-name>
      <scheme-name>ContentBinaryCaches</scheme-name>
    </cache-mapping>
   <cache-mapping>
      <cache-name>repo.ucm.resultOfAQueryCache.*</cache-name>
      <scheme-name>ContentResultOfAQueryCaches</scheme-name>
    </cache-mapping>
   <cache-mapping>
      <cache-name>repo.ucm.contentsUnderAFolderCache.*</cache-name>
      <scheme-name>ContentUnderAFolderCaches</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>ContentNodeCaches</scheme-name>
      <expiry-delay>1m</expiry-delay>
      <high-units>100</high-units>
    </local-scheme>
    <local-scheme>
      <scheme-name>ContentTypeCaches</scheme-name>
      <expiry-delay>30m</expiry-delay>
      <high-units>50</high-units>
    </local-scheme>
    <local-scheme>
      <scheme-name>ContentBinaryCaches</scheme-name>
      <expiry-delay>1m</expiry-delay>
      <high-units>100000</high-units>
      <unit-calculator>
        <class-scheme>
          <class-name>com.tangosol.net.cache.SimpleMemoryCalculator</
class-name>
        </class-scheme>
      </unit-calculator>
    </local-scheme>
    <local-scheme>
      <scheme-name>ContentSearchCaches</scheme-name>
      <expiry-delay>5m</expiry-delay>
      <high-units>50</high-units>
    </local-scheme>
   <local-scheme>
      <scheme-name>ContentSecurityCaches</scheme-name>
      <expiry-delay>10m</expiry-delay>
      <high-units>50</high-units>
    </local-scheme>
   <local-scheme>
```

```
      <scheme-name>ContentProfileCaches</scheme-name>
      <expiry-delay>1h</expiry-delay>
      <high-units>100</high-units>
    </local-scheme>
   <local-scheme>
      <scheme-name>ContentResultOfAQueryCaches</scheme-name>
      <expiry-delay>10m</expiry-delay>
      <high-units>1000</high-units>
    </local-scheme>
  <local-scheme>
      <scheme-name>ContentUnderAFolderCaches</scheme-name>
      <expiry-delay>10m</expiry-delay>
      <high-units>1000</high-units>
    </local-scheme>
    <!--
    <class-scheme>
      <scheme-name>ContentDisabledCaches</scheme-name>
      <class-name>com.tangosol.util.NullImplementation$NullMap</class-name>
    </class-scheme>
    -->
  </caching-schemes>
</cache-config>
```

Configuring the Cache to Check for External Oracle WebCenter Content Server Changes

This section describes how you can change the Content Server's Cache Invalidation Interval so that changes are picked up.

This section includes the following topics:

- Modifying Oracle WebCenter Content Server's Contributor Data Files
- Modifying Oracle WebCenter Content Server's Cache Invalidation Interval
- Testing the Cache Settings

Modifying Oracle WebCenter Content Server's Contributor Data Files

The Content Presenter task flow enables Oracle WebCenter Portal users with Page-Edit permissions to customize the selection and presentation of content. In Content Presenter you can select a single item of content, contents under a folder, a list of items, or a query for content and then select a Content Presenter template with which to render that content on a page in WebCenter Portal.

As well as displaying Oracle WebCenter Content Server folders and files, Content Presenter also integrates with Oracle Site Studio to let you to create, access, edit, and display Site Studio contributor data files (that is, a WebCenter Content Server document) in either a Site Studio region template, or in a custom Content Presenter display template. For more information about creating Content Presenter display templates, see Developing Content Presenter Display Templates in *Developing for Oracle WebCenter Portal*.

In some cases you may want to modify WebCenter Content Server's contributor data files directly through WebCenter Content Server. This operation is completely supported. However, if a contributor data file is being modified through a method other than using

WebCenter Portal, a running WebCenter Portal page that also uses the same data file will not immediately see those updates. This is due to the WebCenter Portal page using Content Presenter to display the contents of the data file while WebCenter Portal is using the cached version of the data file. Fortunately, there is a way to configure the cache so that changes like this are picked up quickly and automatically.

Modifying Oracle WebCenter Content Server's Cache Invalidation Interval

By changing the Oracle WebCenter Content Server's Cache Invalidation Interval, you can enable the cache to be monitored by the cache sweeper utility.

The cache sweeper queries for changes in WebCenter Content Server, flagging the cache as "dirty" if there have been any changes. This causes the application to retrieve a new copy of the document from WebCenter Content Server that replaces the cached version.

By default, the initial value for the Cache Invalidation Interval is set to `0` (minutes). This means that the sweeper has been turned off. To turn the sweeper on, you need to set a value (in minutes). The minimum value that can be set is `2` (minutes). You can do this from the Cache Details page in Fusion Middleware Control or using a WLST command.

This section includes the following topics:

- Modifying the Cache Invalidation Interval Using Fusion Middleware Control
- Modifying the Cache Invalidation Interval Using WLST

Modifying the Cache Invalidation Interval Using Fusion Middleware Control

You can change the Cache Invalidation Interval using Fusion Middleware Control.

To change the Cache Invalidation Interval using Fusion Middleware Control:

1. Log in to Fusion Middleware Control and navigate to the home page for Oracle WebCenter Portal.
2. From the WebCenter Portal menu, select **Settings** and then **Service Configuration**.
3. On the WebCenter Portal Services Configuration page, from the list of services select **Content Repository**.
4. Select the connection name and click **Edit**.
5. In the Cache Details section, set the **Cache Invalidation Interval** to `2` (the shortest time allowed) or a similarly low value.

> **Note:**
>
> In some instances, once the value of the Cache Invalidation Interval has been set (and saved) in Fusion Middleware Control, it becomes sticky and the interval value can only be set back to `0` using the `setContentServerConnection` WLST command.

Modifying the Cache Invalidation Interval Using WLST

You can update the value for the Cache Invalidation Interval using the `setContentServerConnection` WLST command.

Run the command as follows:

```
setContentServerConnection(appName, name, [socketType, url, serverHost, serverPort,
keystoreLocation, keystorePassword, privateKeyAlias, privateKeyPassword,
webContextRoot, clientSecurityPolicy, cacheInvalidationInterval,
binaryCacheMaxEntrySize, adminUsername, adminPassword, extAppId, timeout, isPrimary,
server, applicationVersion])
```

For example:

```
setContentServerConnection(appName='webcenter',name='UCM',
socketType='socket', serverHost='webcenter.oracle.local', serverPort='4444',
webContextRoot='/cs', cacheInvalidationInterval='2',
binaryCacheMaxEntrySize='1024',adminUsername='wccadmin',isPrimary=1)
```

> **Tip:**
>
> To get the other parameter values required to execute the command, you can use the `listContentServerConnections(appName='webcenter',verbose=true)` command.

> **Note:**
>
> You must restart the Oracle WebCenter Portal managed server (`WC_Portal`) for the change to take effect.

Testing the Cache Settings

Once the sweeper is turned on, only cache objects that have been changed will be invalidated.

To test this out, configure Oracle WebCenter Content Server so that it monitors and reports on events.

To configure Oracle WebCenter Content Server to monitor and report on events:

1. Log in to the WebCenter Content Server console application, and under the Administration menu item, select System Audit Information.

   If your console is using the left menu display option, the Administration link will be located there.

2. Under the Tracing Sections Information, add in only `system` and `requestaudit` in the Active Sections. Check Full Verbose Tracing, check Save, then click the Update button. Once this is done, select the View Server Output menu option. This will change the

browser view to display the log. This is all that is needed to configure WebCenter Content Server.

For example, the following is the View Server Output with the cache invalidation interval set to 2 (minutes) Note the time stamp:

```
requestaudit/6 08.30 09:52:26.001  IdcServer-68    GET_FOLDER_HISTORY_REPORT
[dUser=wccadmin][IsJava=1] 0.016933999955654144(secs)
requestaudit/6 08.30 09:52:26.010  IdcServer-69    GET_FOLDER_HISTORY_REPORT
[dUser=wccadmin][IsJava=1] 0.006134999915957451(secs)
requestaudit/6 08.30 09:52:26.014  IdcServer-70
GET_DOCUMENT_HISTORY_REPORT [dUser=wccadmin][IsJava=1]
0.004271999932825565(secs)

... other trace info ...

requestaudit/6 08.30 09:54:26.002  IdcServer-71    GET_FOLDER_HISTORY_REPORT
[dUser=wccadmin][IsJava=1] 0.020323999226093292(secs)
requestaudit/6 08.30 09:54:26.011  IdcServer-72    GET_FOLDER_HISTORY_REPORT
[dUser=wccadmin][IsJava=1] 0.017928000539541245(secs)
requestaudit/6 08.30 09:54:26.017  IdcServer-73
GET_DOCUMENT_HISTORY_REPORT [dUser=wccadmin][IsJava=1]
0.010185999795794487(secs)
```

3. Once the tracing logs are reporting correctly, the next step is set up Oracle WebCenter Portal to test the sweeper. You can do this by setting up two pages with Content Presenter task flows, with each task flow using a different custom Content Presenter display template, and assigning each page a different contributor data file (document in the cache).

   When the WebCenter Portal pages containing the content is loaded in the browser for the first time, you can see the tracing information in the Content Server output viewer. For example:

```
requestaudit/6 08.30 11:51:12.030 IdcServer-129 CLEAR_SERVER_OUTPUT
[dUser=weblogic] 0.029171999543905258(secs)
requestaudit/6 08.30 11:51:12.101 IdcServer-130 GET_SERVER_OUTPUT
[dUser=weblogic] 0.025721000507473946(secs)
requestaudit/6 08.30 11:51:26.592 IdcServer-131 VCR_GET_DOCUMENT_BY_NAME
[dID=919][dDocName=DF_UCMCACHETESTER]
[dDocTitle=DF_UCMCacheTester][dUser=weblogic]
[RevisionSelectionMethod=LatestReleased][IsJava=1] 0.21525299549102783(secs)
requestaudit/6 08.30 11:51:27.117 IdcServer-132 VCR_GET_CONTENT_TYPES
[dUser=wccadmin][IsJava=1] 0.5059549808502197(secs)
requestaudit/6 08.30 11:51:27.146 IdcServer-133 VCR_GET_CONTENT_TYPE
[dUser=wccadmin][IsJava=1] 0.03360399976372719(secs)
requestaudit/6 08.30 11:51:27.169 IdcServer-134 VCR_GET_CONTENT_TYPE
[dUser=wccadmin][IsJava=1] 0.008806000463664532(secs)
requestaudit/6 08.30 11:51:27.204 IdcServer-135 VCR_GET_CONTENT_TYPE
[dUser=wccadmin][IsJava=1] 0.013265999965369701(secs)
requestaudit/6 08.30 11:51:27.384 IdcServer-136 VCR_GET_CONTENT_TYPE
[dUser=wccadmin][IsJava=1] 0.18119299411773682(secs)
requestaudit/6 08.30 11:51:27.533 IdcServer-137 VCR_GET_CONTENT_TYPE
[dUser=wccadmin][IsJava=1] 0.1519480049610138(secs)
requestaudit/6 08.30 11:51:27.634 IdcServer-138 VCR_GET_CONTENT_TYPE
[dUser=wccadmin][IsJava=1] 0.10827399790287018(secs)
requestaudit/6 08.30 11:51:27.687 IdcServer-139 VCR_GET_CONTENT_TYPE
[dUser=wccadmin][IsJava=1] 0.059702999889850616(secs)
requestaudit/6 08.30 11:51:28.271 IdcServer-140 GET_USER_PERMISSIONS
[dUser=weblogic][IsJava=1] 0.006703000050038099(secs)
requestaudit/6 08.30 11:51:28.285 IdcServer-141 GET_ENVIRONMENT
[dUser=wccadmin][IsJava=1] 0.010893999598920345(secs)
```

```
requestaudit/6 08.30 11:51:30.433 IdcServer-142 GET_SERVER_OUTPUT [dUser=weblogic]
0.017318999394774437(secs)
requestaudit/6 08.30 11:51:41.837 IdcServer-143 VCR_GET_DOCUMENT_BY_NAME [dID=508]
[dDocName=113_ES]
[dDocTitle=Landing Home][dUser=weblogic][RevisionSelectionMethod=LatestReleased]
[IsJava=1] 0.15937699377536774(secs)
requestaudit/6 08.30 11:51:42.781 IdcServer-144 GET_FILE [dID=326]
[dDocName=WEBCENTERORACL000315][dDocTitle=Duke][dUser=anonymous]
[RevisionSelectionMethod=LatestReleased][dSecurityGroup=Public][xCollectionID=0]
0.16288499534130096(secs)
```

The highlighted sections show where the two example data files DF_UCMCACHETESTER and 113_ES were called by the WebCenter Portal VCR connection to WebCenter Content Server. Note the VCR_GET_DOCUMENT_BY_NAME invocation.

On subsequent refreshes of these two pages, you will notice (after you refresh WebCenter Content Server's View Server Output) that there are no further traces of the same VCR_GET_DOCUMENT_BY_NAME invocations. This is because the pages are getting the documents from the cache.

4. The next step is to go through the back door and change one of the documents through the Content Server console. To do this, locate the data file document, and from the Content Information page, select **Edit Data File**.

   This invokes the Site Studio Contributor, where you can make some modifications.

   When you refresh the Content Server View Server Output, the tracing displays the operations performed on the document.

```
requestaudit/6 08.30 11:56:59.972 IdcServer-255 SS_CHECKOUT_BY_NAME [dID=922]
[dDocName=DF_UCMCACHETESTER][dUser=weblogic]
[dSecurityGroup=Public] 0.05558200180530548(secs)
requestaudit/6 08.30 11:57:00.065 IdcServer-256 SS_GET_CONTRIBUTOR_CONFIG [dID=922]
[dDocName=DF_UCMCACHETESTER]
[dDocTitle=DF_UCMCacheTester][dUser=weblogic][dSecurityGroup=Public]
[xCollectionID=0] 0.08632399886846542(secs)
requestaudit/6 08.30 11:57:00.470 IdcServer-259 DOC_INFO_BY_NAME [dID=922]
[dDocName=DF_UCMCACHETESTER]
[dDocTitle=DF_UCMCacheTester][dUser=weblogic][dSecurityGroup=Public]
[xCollectionID=0] 0.02268899977207184(secs)
requestaudit/6 08.30 11:57:10.177 IdcServer-264 GET_FOLDER_HISTORY_REPORT
[dUser=wccadmin][IsJava=1] 0.007652000058442354(secs)
requestaudit/6 08.30 11:57:10.181 IdcServer-263 GET_FOLDER_HISTORY_REPORT
[dUser=wccadmin][IsJava=1] 0.01868399977684021(secs)
requestaudit/6 08.30 11:57:10.187 IdcServer-265 GET_DOCUMENT_HISTORY_REPORT
[dUser=wccadmin][IsJava=1] 0.009367000311613083(secs)
(internal)/6 08.30 11:57:26.118 IdcServer-266 File to be removed: /oracle/app/
admin/domains/webcenter/ucm/cs/vault/~temp/703253295.xml
(internal)/6 08.30 11:57:26.121 IdcServer-266 File to be removed: /oracle/app/
admin/domains/webcenter/ucm/cs/vault/~temp/703253295.xml
requestaudit/6 08.30 11:57:26.122 IdcServer-266 SS_SET_ELEMENT_DATA [dID=923]
[dDocName=DF_UCMCACHETESTER]
[dDocTitle=DF_UCMCacheTester][dUser=weblogic][dSecurityGroup=Public]
[xCollectionID=0][StatusCode=0][StatusMessage=Successfully checked in content item
'DF_UCMCACHETESTER'.] 0.3765290081501007(secs)
requestaudit/6 08.30 11:57:30.710 IdcServer-267 DOC_INFO_BY_NAME [dID=923]
[dDocName=DF_UCMCACHETESTER]
[dDocTitle=DF_UCMCacheTester][dUser=weblogic][dSecurityGroup=Public]
[xCollectionID=0] 0.07942699640989304(secs)
requestaudit/6 08.30 11:57:30.733 IdcServer-268 SS_GET_CONTRIBUTOR_STRINGS
[dUser=weblogic] 0.0044570001773536205(secs)
```

After refreshing the first page, you should see that the updates have been applied. Note that the refresh time may vary since the Cache Invalidation Interval (set to 2 minutes) is not determined by when changes occur. The sweeper just runs every two minutes.

When you refresh the WebCenter Content Server View Server Output, for this example, the tracing displays the following information:

```
requestaudit/6 08.30 11:59:10.171 IdcServer-270 GET_FOLDER_HISTORY_REPORT
[dUser=wccadmin][IsJava=1] 0.00952600035816431(secs)
requestaudit/6 08.30 11:59:10.179 IdcServer-271 GET_FOLDER_HISTORY_REPORT
[dUser=wccadmin][IsJava=1] 0.011118999682366848(secs)
requestaudit/6 08.30 11:59:10.182 IdcServer-272 GET_DOCUMENT_HISTORY_REPORT
[dUser=wccadmin][IsJava=1] 0.007447000127285719(secs)
requestaudit/6 08.30 11:59:16.885 IdcServer-273 VCR_GET_DOCUMENT_BY_NAME
[dID=923][dDocName=DF_UCMCACHETESTER]
[dDocTitle=DF_UCMCacheTester][dUser=weblogic]
[RevisionSelectionMethod=LatestReleased][IsJava=1] 0.0786449983716011(secs)
```

After the specified Cache Invalidation Interval time, the sweeper is invoked (tracked by the `GET_` calls). Since a change has been noted, the next call is to the `VCR_GET_DOCUMENT_BY_NAME` to retrieve a new version of the modified data file.

Navigating back to the second page and viewing the server output, there are no further `VCR_GET_DOCUMENT_BY_NAME` to retrieve the data file. This simply means that the data file was just retrieved from the cache. Looking at the example server output, we can see that there was only one request for the `VCR_GET_DOCUMENT_BY_NAME`:

```
requestaudit/6 08.30 12:08:00.021 Audit Request Monitor Request Audit Report
over the last 120 Seconds for server webcenteroraclelocal16200****
requestaudit/6 08.30 12:08:00.021 Audit Request Monitor -Num Requests 8
Errors 0 Reqs/sec. 0.06666944175958633
Avg. Latency (secs) 0.02762500010430813 Max Thread Count 2
requestaudit/6 08.30 12:08:00.021 Audit Request Monitor 1 Service
VCR_GET_DOCUMENT_BY_NAME
Total Elapsed Time (secs) 0.09200000017881393 Num requests 1 Num errors 0
Avg. Latency (secs) 0.09200000017881393
requestaudit/6 08.30 12:08:00.021 Audit Request Monitor 2 Service
GET_PERSONALIZED_JAVASCRIPT
Total Elapsed Time (secs) 0.054999999701976776 Num requests 1 Num errors 0
Avg. Latency (secs) 0.054999999701976776
requestaudit/6 08.30 12:08:00.021 Audit Request Monitor 3 Service
GET_FOLDER_HISTORY_REPORT
Total Elapsed Time (secs) 0.028999999165534973 Num requests 2 Num errors 0
Avg. Latency (secs) 0.014499999582767487
requestaudit/6 08.30 12:08:00.021 Audit Request Monitor 4 Service
GET_SERVER_OUTPUT
Total Elapsed Time (secs) 0.017999999225139618 Num requests 1 Num errors 0
Avg. Latency (secs) 0.017999999225139618
requestaudit/6 08.30 12:08:00.021 Audit Request Monitor 5 Service GET_FILE
Total Elapsed Time (secs) 0.013000000268220901 Num requests 1 Num errors 0
Avg. Latency (secs) 0.013000000268220901
requestaudit/6 08.30 12:08:00.021 Audit Request Monitor ****End Audit
Report*****
```

Deleting Oracle WebCenter Content Server Connections

This section includes the following topics:

- Deleting Oracle WebCenter Content Server Connections Using Fusion Middleware Control
- Deleting Oracle WebCenter Content Server Connections Using WLST

> **Note:**
>
> Delete a WebCenter Content Server connection only if it is not in use. If a connection is marked as the default connection, it should first be removed from the active list, and then deleted.

Deleting Oracle WebCenter Content Server Connections Using Fusion Middleware Control

You can delete an Oracle WebCenter Content Server connection using Fusion Middleware Control.

To delete a content repository connection:

1. Log in to Fusion Middleware Control and navigate to the home page for Oracle WebCenter Portal.
2. From the WebCenter Portal menu, select **Settings** and then **Service Configuration**.
3. On the WebCenter Portal Services Configuration page, from the list of services 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 is deployed.

Deleting Oracle WebCenter Content Server Connections Using WLST

Use the WLST command `deleteContentServerConnection` to remove a content repository connection.

For command syntax and examples, see deleteContentServerConnection in *WebCenter WLST Command Reference Reference*.

> **Note:**
>
> To effect this change you must restart the managed server on which Oracle WebCenter Portal is deployed.

Changing the Maximum File Upload Size

You can specify the maximum upload size for files.

For files uploaded from features such as a wiki or blog, the maximum file upload size is 2 GB. For information about changing the maximum upload size, see webcenter-config.xml.

The maximum upload size for files uploaded using Content Manager is 50 MB. For information about changing the maximum upload size, see Modifying the File Upload Size in Content Manager.

Configuring Content Manager for Oracle Content and Experience Cloud

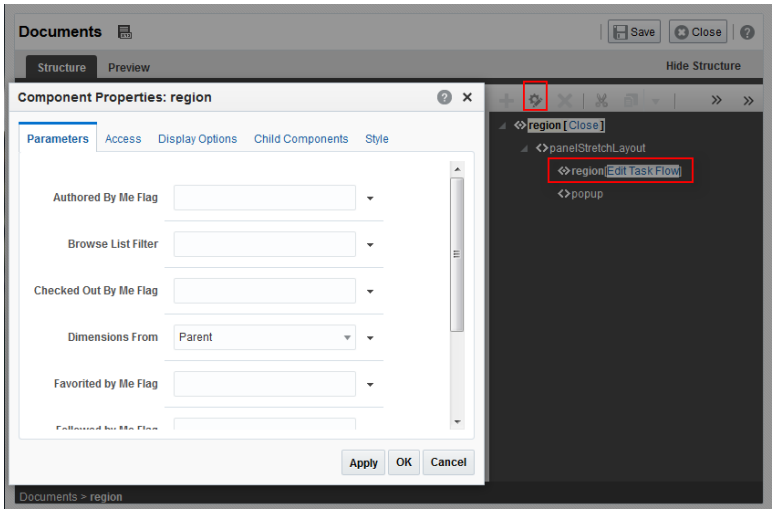
Content Manager enables you to upload files to Content Server, the content repository for WebCenter Portal. Files are organized into document libraries and folders. Depending on your permissions, you can open, edit, delete, copy, rename, move, share, search, view, and manage information about files and work with libraries and folders in the connected content repository.

Content Manager supports Hybrid Enterprise Content Management (HECM), which helps portal members to easily and rapidly access enterprise content in Oracle Content and Experience Cloud.

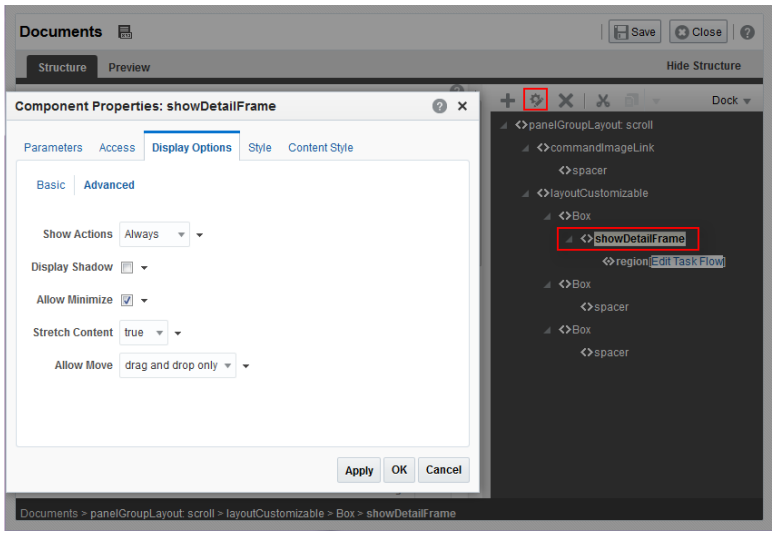
To enable Content Manager in WebCenter Portal to access Oracle Content and Experience Cloud, you must first configure Content Server to integrate with Oracle Content and Experience Cloud. See Configuring Document Cloud Service Integration Settings in *Administering Oracle WebCenter Content*.

To view Oracle Content and Experience Cloud content in Content Manager, you will need to customize properties in Structure view to allow for page stretching:

1. In WebCenter Portal administration, go to the **System Pages** page.
2. For the **Documents** page (which exposes Content Manager), click the **Customize** link.
3. In Structure view, select the `region[Edit Task Flow]` node, and click the **Show the properties of region** icon.



4. On the **Parameters** tab, set **Dimensions From** to `Parent`.
5. In Structure view, select the `showDetailFrame` node (the task flow container), and click the **Show the properties of showDetailFrame** icon.



6. In the Component Properties dialog, set either of the following:
   - On the **Display Options** tab, set **Stretch Content** to `true`.
   - On the **Content Style** tab, set **Height** as required. For example, `800px`.

7

Managing Analytics

Configure and manage Analytics in WebCenter Portal to display usage and performance metrics for a portal.
Always use Fusion Middleware Control or the WLST command-line tool to review and configure back-end services for WebCenter Portal. Any configuration changes that you make post-deployment are stored in the MDS metadata store as customizations. 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.

> **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 granted through WebCenter Portal Administration.
>
> For more information about roles and permissions, see Understanding Administrative Operations, Roles, and Tools.

Topics:

- About Analytics in WebCenter Portal
- Configuration Roadmap for Analytics
- Analytics Prerequisites
- Configuring Analytics Collector Settings
- Registering an Analytics Collector for Your Application
- Validating Analytic Event Collection
- Viewing the Current WebCenter Portal's Analytic Event List
- Purging Analytics Data
- Partitioning Analytics Data

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, and portlet 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 information, see Query Options in *Building Portals with Oracle WebCenter Portal*.

An overview of Analytics components and ready-to-use task flows are described in the following sections:

- Analytics Components
- Analytics Task Flows

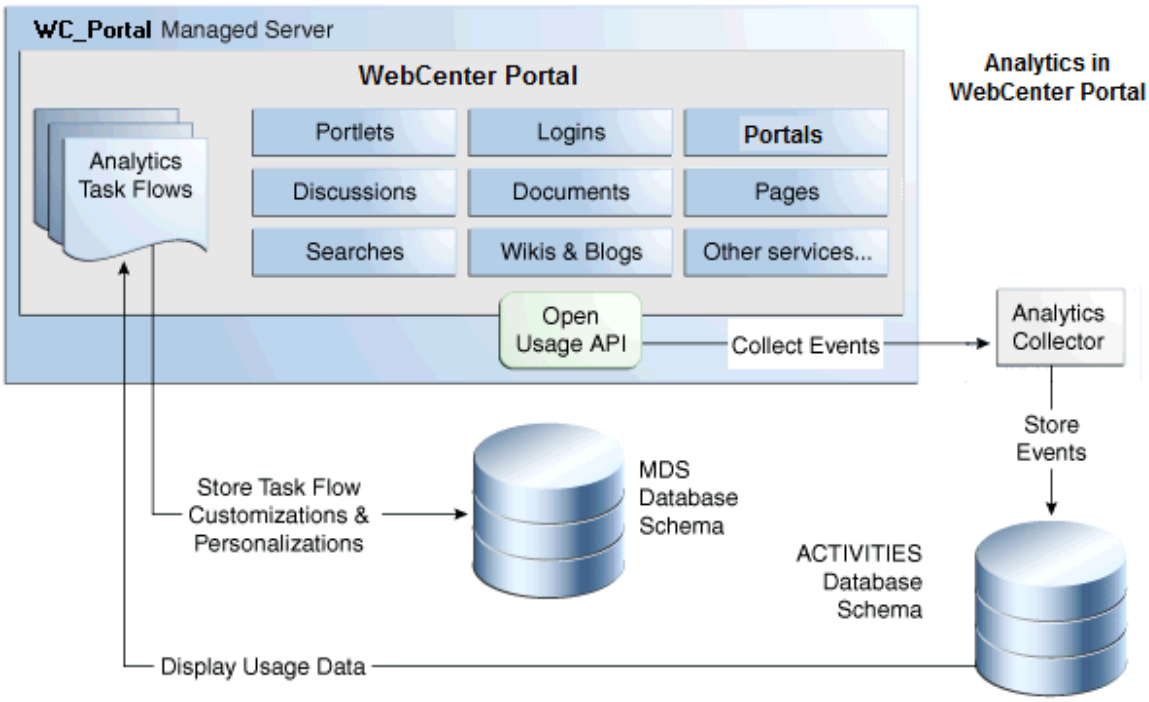
Analytics Components

The following figure illustrates components for Analytics in WebCenter Portal:

- **`WC_Portal`** – The managed server on which Oracle WebCenter Portal and the Analytics Collector is deployed are 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 Metadata Service (MDS) repository that stores task flow customizations.

Figure 7-1 Analytics Components



Analytics Task Flows

Table 7-1 lists the Analytics task flows available with WebCenter Portal. For detailed information about these task flows and how to use them, see About Analytics in *Building Portals with Oracle WebCenter Portal*.

Table 7-1 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.
Wiki Metrics	Tracks most popular/least popular wikis.
Blog Metrics	Tracks most popular/least popular blogs.

Table 7-1 (Cont.) Analytics Task Flows in WebCenter Portal

Analytics Task Flows	Description
Discussion Metrics	Tracks most popular/least popular discussions.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

Portal Traffic	Displays usage data for a portal.
Portal Response Time	Displays page performance data for a portal.

Configuration Roadmap for Analytics

The flow chart in Figure 7-2 and tasks in Table 7-2 provide an overview of the prerequisites and tasks required to get Analytics working in WebCenter Portal.

Figure 7-2 Configuring Analytics for Use in WebCenter Portal

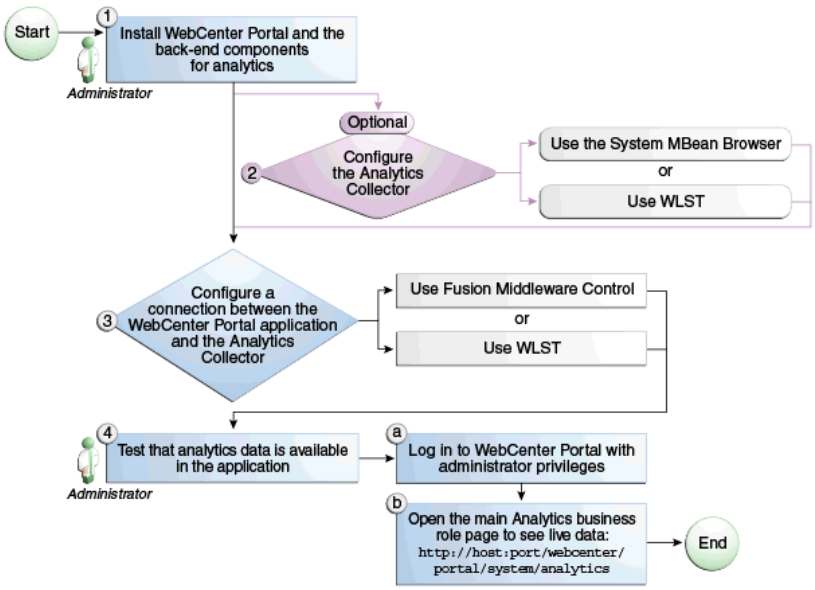


Table 7-2 Configuring Analytics for Use in WebCenter Portal

Actor	Task	Link
Administrator	**1.** Install Oracle WebCenter Portal and the Oracle WebCenter Portal Analytics Collector component	See About Oracle WebCenter Portal Installation in *Installing and Configuring Oracle WebCenter Portal*
Administrator	**2.** (Optional) Configure Analytics Collector settings using either of the following tools: • Fusion Middleware Control • WLST	See Configure Analytics Collector Settings
Administrator	**3.** Configure a connection between the Oracle WebCenter Portal and the Analytics Collector using either of the following tools: • Fusion Middleware Control • WLST	See Registering an Analytics Collector for Your Application
WebCenter Portal Administrator	**4.** Test that analytics data is available in WebCenter Portal • **4.a** Log in to WebCenter Portalwith administrator privileges • **4.b** Open the main Analytics business role page to see live data: `http://host:port/webcenter/portal/system/Analytics`	

Analytics Prerequisites

This section includes the following topics:

- Analytics – Installation
- Analytics – Configuration
- Analytics – Security Considerations
- Analytics – Limitations

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 About Oracle WebCenter Portal Installation in *Installing and Configuring Oracle WebCenter Portal*.

The Analytics schema (`ACTIVITIES`) and the WebCenter Portal schema (`WEBCENTER`) can be installed on the same database or on separate databases.

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 information, see 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 you must register the Analytics Collector and enable event collection. For more information, see 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.

Analytics – Security Considerations

In WebCenter Portal, resource catalogs display Analytics task flows only to users with appropriate permissions:

- Administrators – Users with the `Administrator` role have access to all Analytics task flows
- Portal Managers – Within a particular portal, members with the `Portal Manager` 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 managers must manage page security such that only appropriate, specified users have access to pages that expose analytics data. See About Analytics in *Building Portals with Oracle WebCenter Portal*.

Analytics – Limitations

Analytics task flows do not display custom event information.

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** - Unicast
- **Clustering** - The clustering settings do not apply. Clustering is not supported in this version.

> **Note:**
>
> If the database used by WebCenter Portal uses a National Character Set set to something other than `AL16UTF16`, the Analytics startup listener may fail to start. The National Character Set option is configure through the Database Configuration Assistant when the database is created. Oracle recommends that you keep the National Character Set set to its default value of `AL16UTF16` to avoid potential issues.

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`).

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 setAnalyticsCollectorConfig in *WebCenter WLST Command Reference Reference*.

> **Note:**
>
> To start using the property values you must restart the managed server on which the Analytics Collector application is deployed (`WC_Portal`).

Setting Analytics Collector Properties Using Fusion Middleware Control

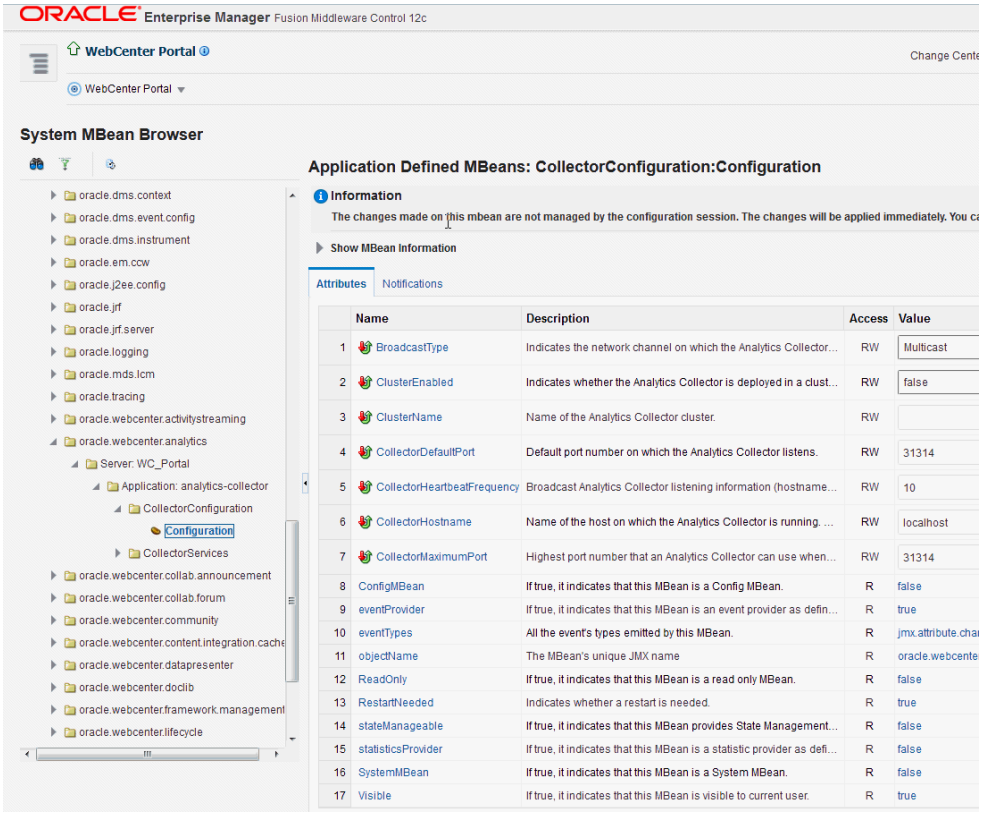
Use the Systems MBeans Browser in Fusion Middleware Control to set event collection properties for the Analytics Collector:

To configure the Analytics Collector (deployed on the `WC_Portal` managed server):

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal.
2. Open the System MBean Browser. From the **WebCenter Portal** menu, select **System MBean Browser**.
3. Navigate to: **Application Defined MBeans >oracle.webcenter.analytics >Server: `WC_Portal` >Application: analytics-collector >CollectorConfiguration >Configuration**

   Alternatively, search for `CollectorConfiguration` or filter the System MBean Browser tree using the MBean pattern: `oracle.webcenter.analytics:*`

Figure 7-3 System MBeans Browser - Analytics Collector Properties



4. Modify configuration properties for the Analytics Collector.

Table 7-3 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.
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.
CollectorHeartbeatFrequency	The clustering settings do not apply. Clustering is not supported in this version.
CollectorDefaultPort	Enter the default port number on which the Analytics Collector listens. The default value is `31314`.
CollectorHostName	Enter the name of the host on which the Analytics Collector is running. The default setting is `localhost`.

Table 7-3 (Cont.) Analytics Collector - Configuration Properties

Field	Description
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.

5. To start using the new settings restart the managed server on which the Analytics Collector application is deployed (`WC_Portal`).

Registering an Analytics Collector for Your Application

Events raised in WebCenter Portal using OpenUsage APIs can be sent to an Analytics Collector for use by Analytics. If you intend to use any of the features or task flows provided by these tools you must connect WebCenter Portal to an Analytics Collector.

While you can register multiple Analytics Collector connections for WebCenter Portal, only one Analytics Collector is used (i.e., the default (or active) connection).

To start using a new configuration you must restart the managed server on which WebCenter Portal is deployed.

This section includes the following subsections:

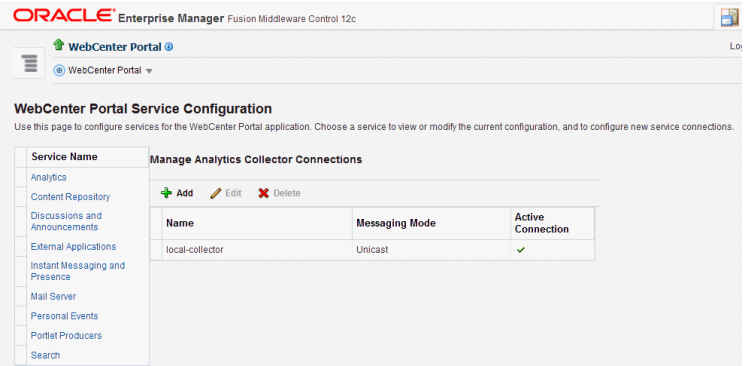
- Registering an Analytics Collector Using Fusion Middleware Control
- Registering an Analytics Collector Using WLST
- Disabling WebCenter Portal Event Collection

Registering an Analytics Collector Using Fusion Middleware Control

To register an Analytics Collector for WebCenter Portal:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal.
2. Open the Service Configuration page. From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
3. From the list of services on the WebCenter Portal Service Configuration page, select **Analytics**.
4. To connect to an Analytics Collector, click **Add** (Figure 7-4).

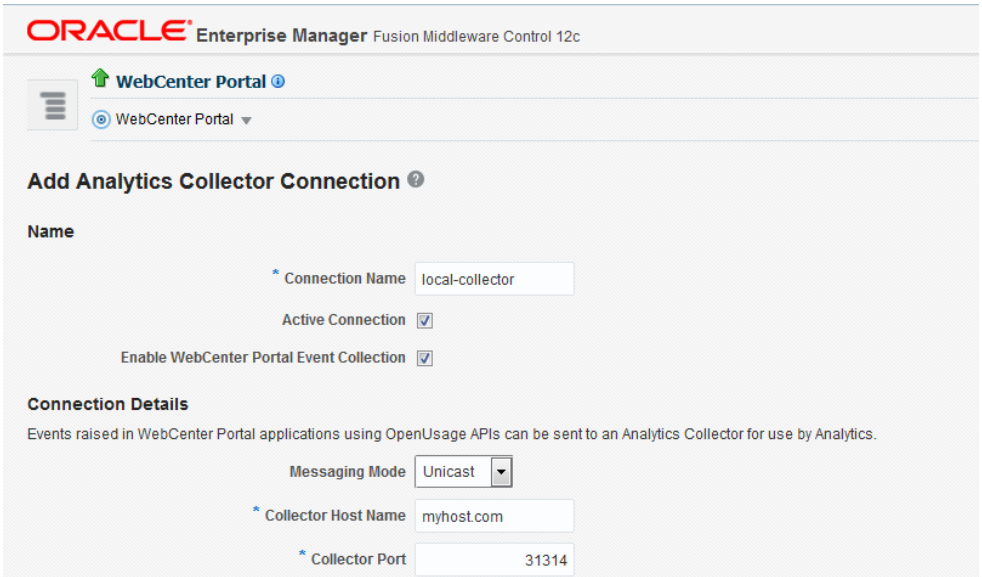
Figure 7-4 Configuring Analytics Collector Connections



5. Enter a unique name for this connection.

   The name must be unique (across all connection types) within WebCenter Portal.

Figure 7-5 Add Analytics Collector Connection



6. Select **Active Connection** to use this connection for Analytics.

   While you can register multiple Analytics Collector connections for WebCenter Portal, 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.

Table 7-4 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`.

9. Click **OK** to save.
10. To start using the new (active) connection you must restart the managed server on which WebCenter Portal is deployed.

Registering an Analytics Collector Using WLST

Use the WLST command `createAnalyticsCollectorConnection` to create an Analytics Collector connection for WebCenter Portal. To update an existing connection, use `setAnalyticsCollectorConnection`. For command syntax and examples, see createAnalyticsCollectorConnection and setAnalyticsCollectorConnection in *WebCenter WLST Command Reference Reference*.

> **Note:**
>
> To start using the new connection, ensure that `isEnabled=1` and `default=1`, and then restart the managed server on which WebCenter Portal is deployed.

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:

- Disabling WebCenter Portal Event Collection Using Fusion Middleware Control
- Disabling WebCenter Portal Event Collection Using WLST

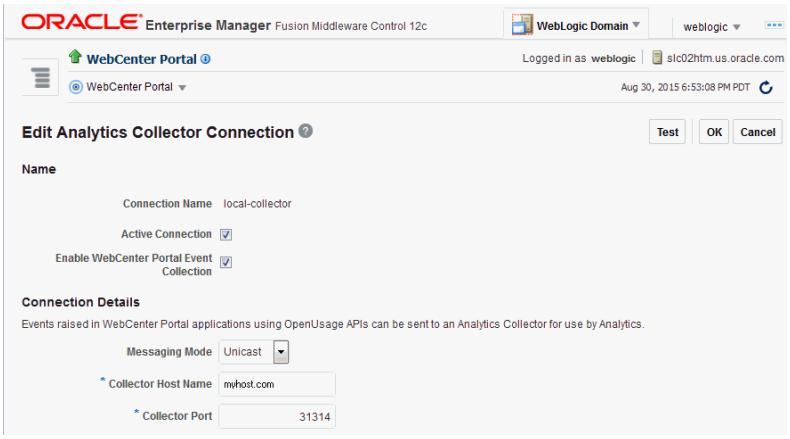
Disabling WebCenter Portal Event Collection Using Fusion Middleware Control

To disable event collection for WebCenter Portal:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal.
2. Open the Service Configuration page. From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.

3. From the list of services on the WebCenter Portal Service Configuration page, select **Analytics**.
4. Select the connection in the table, and then click **Edit**.
5. Deselect **Enable WebCenter Portal Event Collection** (Figure 7-6).

Figure 7-6 Disabling Analytics Event Collection



6. To effect this change you must restart the managed server on which WebCenter Portal is deployed.

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 setAnalyticsCollectorConnection in *WebCenter WLST Command Reference Reference*.

Validating Analytic Event Collection

You can check whether events reach the Analytics Collector by checking the trace log as follows:

1. Set the log level for the logger `com.plumtree.analytics` to FINEST.
2. After this change, the events reaching the Analytics Collector will be the service log file, `[DOMAIN_HOME]/servers/WC_Portal/logs/analytics-collector/service.log`.
   Event messages are similar to the following:

```
[2021-02-26T02:27:44.327-08:00] [WC_Portal] [TRACE] []
 [com.plumtree.analytics.collector.AnalyticsPmbListenerWrapper.eventDump]
 [tid: PMB Message Processor] [ecid:
 d7eccb85-4094-4782-9a7d-6c74f7d96574-00000095,0] [APP: analytics-collector]
 [partition-name: DOMAIN] [tenant-name: GLOBAL] [SRC_CLASS:
 com.plumtree.analytics.logging.Log] [SRC_METHOD: fine] Event = [[
 EVENT_TYPE: {HTTP://WWW.ORACLE.COM/ANALYTICS/WC}LOGINS
 VERSION: 3.0.XXXX
 AS_DIMENSION_USER.USERID: monty
 CLIENT_IP.IP: 10.191.255.82
 USER_AGENT.AGENT: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:78.0)
 Gecko/20100101 Firefox/78.0
```

```
APPLICATION.NAME: webcenter
PROPERTY_VERSION: 3.0.XXXX
REFERRER.URL:
http://den00stq.us.oracle.com:8888/webcenter/system/welcome.jsp?wc.contentSource=
SESSION_ID: 376590077
]]
```

To display analytics collector configuration information, enter the following URL:

`http://`*`hostname`*`:`*`WC_Portal_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)

Viewing the Current WebCenter Portal's Analytic Event List

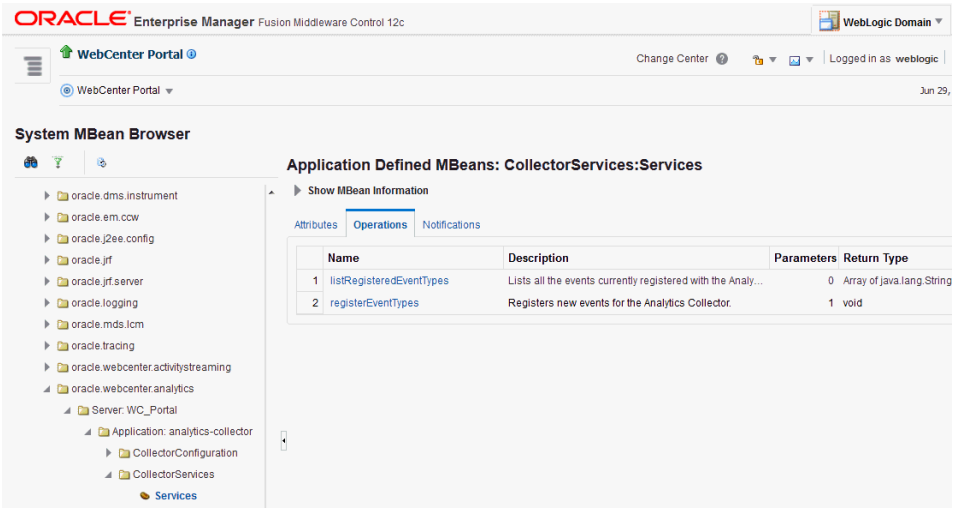
Use the Systems MBeans Browser in Fusion Middleware Control to see which events an Analytics Collector is configured to collect.

To display the current list of analytics events:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal.
2. Open the System MBean Browser. From the **WebCenter Portal** menu, select **System MBean Browser**.
3. Navigate to: **Application Defined MBeans> oracle.webcenter.analytics > Server: `WC_Portal`> 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 7-7 System MBeans Browser - Register Analytics Events



5. Click **listRegisteredEventTypes**.
6. Click **Invoke**.

Alternatively, use the WLST command `listAnalyticsEventTypes`. For command syntax and examples, see listAnalyticsEventTypes in *WebCenter WLST Command Reference Reference*.

Purging Analytics Data

For information about purging analytics data, see Purging Oracle WebCenter Portal's Analytics Data in *Administering Oracle Fusion Middleware*.

Partitioning Analytics Data

For information about partitioning analytics data, see Partitioning Oracle WebCenter Portal's Analytics Data in *Administering Oracle Fusion Middleware*.

8

Managing Calendar Events

Configure and manage events to expose personal Microsoft Exchange calendars inWebCenter Portal portals.
Always use Fusion Middleware Control or the WLST command-line tool to review and configure back-end services for WebCenter Portal. Any configuration changes that you make, post deployment, are stored in the MDS metadata store as customizations.

> **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 is deployed for your changes to take effect.

> **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 granted through WebCenter Portal Administration.
>
> For more information about roles and permissions, see Understanding Administrative Operations, Roles, and Tools.
>
> For troubleshooting, see Troubleshooting Issues with Events.

Topics:

- About Events Connections
- Configuring Personal Events for WebCenter Portal
- Events Prerequisites for Personal Events
- Registering Events Servers
- Choosing the Active Events Server Connection
- Modifying Events Server Connection Details
- Deleting Event Server Connections

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.

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.

Configuring Personal Events for WebCenter Portal

Use the roadmaps in this section as a guide through the configuration process for providing access to personal events:

The flow chart (Figure 8-1) and table (Table 8-1) in this section provide an overview of the prerequisites and tasks required for personal events to work in WebCenter Portal.

Figure 8-1 Configuring Personal Events for WebCenter Portal

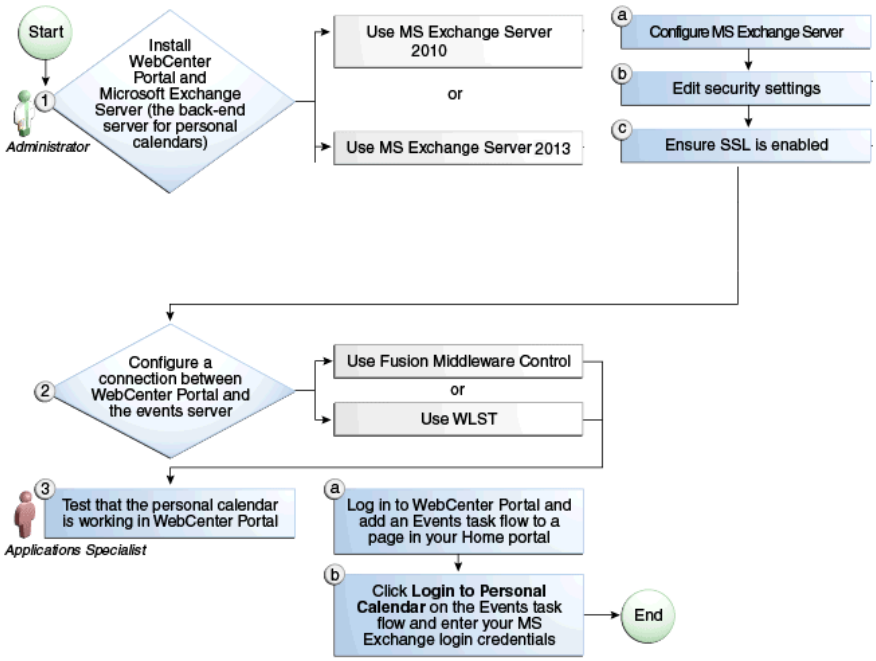


Table 8-1 Configuring the Personal Events for WebCenter Portal

Actor	Task	Subtask	Link
Administrator	**1.** Install WebCenter Portal		See About Oracle WebCenter Portal Installation in *Installing and Configuring Oracle WebCenter Portal*

Table 8-1 (Cont.) Configuring the Personal Events for WebCenter Portal

Actor	Task	Subtask	Link
Administrator	**2.**Install and configure Microsoft Exchange Server release 2013 or 2010. • Install Microsoft Exchange Server 2013	**2.a** Configure MS Exchange Server 2013 **2.b** Edit security settings **2.c** Ensure SSL is enabled	See Microsoft Exchange Server 2013 Prerequisites
Administrator	• Install Microsoft Exchange Server 2010	**2.a** Configure MS Exchange Server 2010 **2.b** Edit security settings **2.c** Ensure SSL is enabled	See Microsoft Exchange Server 2010 Prerequisites
Administrator	**3.** Configure a connection between WebCenter Portal and the events server using either Fusion Middleware Control or WLST		See Registering Events Servers
End User	**4.** 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 Microsoft Exchange Server login credentials	In *Building Portals with Oracle WebCenter Portal*, see: • Adding an Events Task Flow to a Page • Accessing Your Personal Events

Events Prerequisites for Personal Events

This section includes the following subsections:

- Microsoft Exchange Server 2013 Prerequisites
- Microsoft Exchange Server 2010 Prerequisites

Microsoft Exchange Server 2013 Prerequisites

This section describes the Microsoft Exchange Server 2013 prerequisites when used as the server for personal events.

This section includes the following subsections:

- Microsoft Exchange Server 2013 - Installation
- Microsoft Exchange Server 2013 - Configuration
- Microsoft Exchange Server 2013 - Security Considerations
- Microsoft Exchange Server 2013 - Limitations

Microsoft Exchange Server 2013 - Installation

Refer to the Microsoft Exchange Server 2013 documentation for installation information.

Microsoft Exchange Server 2013 - Configuration

To use Microsoft Exchange Server 2013 as the server for personal events, you must edit the Microsoft Exchange Server 2013 web service WSDL to specify the location of the web service.

To specify the location of the Microsoft Exchange Server 2013 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>
   ```

Microsoft Exchange Server 2013 - Security Considerations

Events includes a Microsoft Exchange Server 2013 adapter that communicates with the Microsoft Exchange Server 2013 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. You must enable Basic authentication. Further, you must enable anonymous access to `Services.wsdl`, `Messages.xsd`, and `Types.xsd` so that JAX-WS can access them to create the service port before committing any web service call. This involves creating a virtual directory and enabling anonymous authentication and disabling Windows authentication.

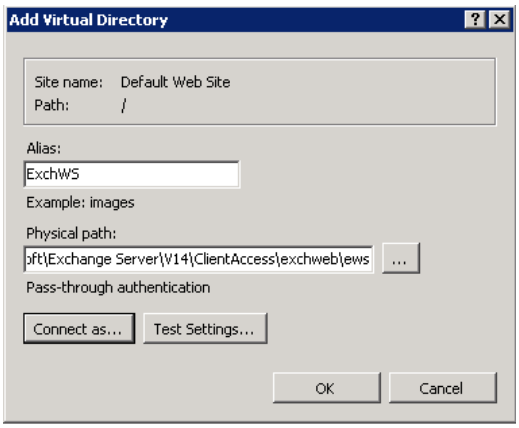
To edit Microsoft Exchange Server security settings:

1. On Microsoft Exchange Server, open Internet Information Services (IIS) Manager.
2. Under **Node *computer_name*** > **Sites > Default Web Site** > **EWS**, double-click **Authentication** under IIS.
3. Right-click **Basic Authentication** and select **Enable** to enable Basic Authentication for the EWS application.
4. Right-click **Anonymous Authentication** and select **Enable** to enable Anonymous Authentication for the EWS application.
5. If IIS version is 7.5, then do the following:
   a. Under Sites, right-click **Default Web Site** and select **Add Virtual Directory** to create a virtual directory that will be used to provide anonymous access to `Services.wsdl`, `Messages.xsd`, and `Types.xsd`.
   b. In the Add Virtual Directory dialog, in the **Alias** field, specify the name of the virtual directory, for example `ExchWS`.

c. In the Physical Path field, specify the path to the virtual directory. For example:

```
C:\Program Files\Microsoft\Exchange Server\V15\ClientAccess\exchweb\ews
```

Figure 8-2 Creating a Virtual Directory



d. Click **Connect as**.

e. In the Connect As dialog, ensure **Application user (pass-thorough authentication)** is selected.

f. Click **OK**.

6. If IIS version is 8.5, then do the following:

a. Under Sites, right-click **Default Web Site** and select **Add Application** to create a application directory that will be used to provide anonymous access to `Services.wsdl`, `Messages.xsd`, and `Types.xsd`.

b. In the Add Application dialog, in the Alias field, specify the name of the directory, for example `ExchWS`. Select Application pool as **defaultApplicationPool**.

c. In the Physical Path field, specify a folder outside of Microsoft Exchange Server installation. For example, `C:\MSExchange`.

d. Copy the web services files `Services.wsdl`, `Messages.xsd`, and `Types.xsd` from the folder `C:\Program Files\Microsoft\Exchange Server\V15\ClientAccess\exchweb\ews` to the folder `C:\MSExchange`.

e. Click **Connect as**.

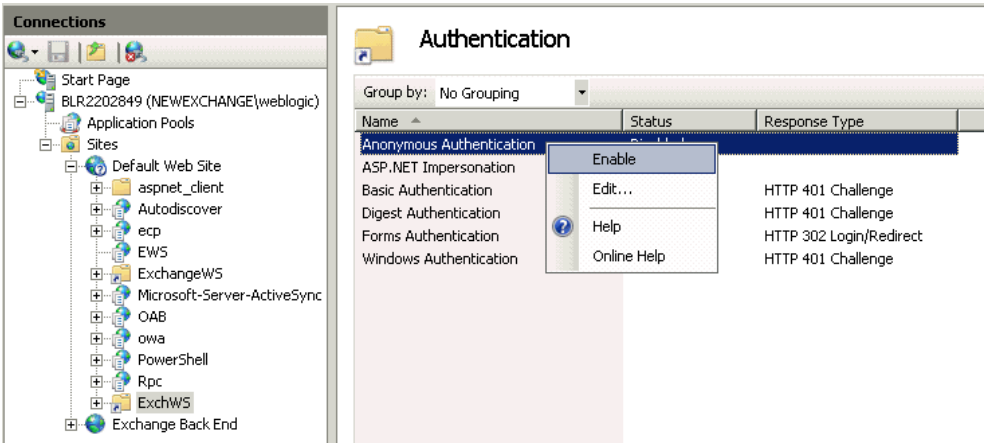
f. In the Connect As dialog, ensure **Application user (pass-thorough authentication)** is selected.

g. Click **OK**.

7. Under **Default Web Site > ExchWS**, double-click **Authentication** under IIS.

8. Right-click **Anonymous Authentication** and select **Enable** (Figure 8-3).

Figure 8-3 Enabling Anonymous Authentication



9. Right-click **Anonymous Authentication** and select **Edit**.
10. In the Edit Anonymous Authentication Credentials dialog, ensure **Application pool identity** is selected.
11. Right-click **Windows Authentication** and select **Disable**.

Events uses Basic Authentication to communicate with the Microsoft Exchange Server. To secure the communication, ensure that SSL is enabled. For more information, see:

http://www.microsoft.com/technet/prodtechnol/WindowsServer2003/Library/IIS/56bdf977-14f8-4867-9c51-34c346d48b04.mspx?mfr=true

You must also ensure that the `SSLAlwaysNegoClientCert` property is set to `true` in IIS. The `SSLAlwaysNegoClientCert` property controls SSL client connection negotiations.

For example, use the following command to set the `SSLAlwaysNegoClientCert` property:

```
CScript.exe adsutil.vbs SET w3svc/1/SSLAlwaysNegoClientCert true
```

For more information about the `SSLAlwaysNegoClientCert` property, see:

http://www.microsoft.com/technet/prodtechnol/WindowsServer2003/Library/IIS/bce0fb87-79ea-40cd-963f-239545b61a12.mspx?mfr=true

For information about setting the SSLAlwaysNegoClientCert property, see:

https://www.microsoft.com/technet/prodtechnol/WindowsServer2003/Library/IIS/d3df4bc9-0954-459a-b5e6-7a8bc462960c.mspx?mfr=true to understand how to use adsutil.vbs

Microsoft Exchange Server 2013 - Limitations

There are currently no known limitations.

Microsoft Exchange Server 2010 Prerequisites

This section describes the Microsoft Exchange Server 2010 prerequisites when used as the server for personal events.

This section includes the following subsections:

- Microsoft Exchange Server 2010 - Installation
- Microsoft Exchange Server 2010 - Configuration
- Microsoft Exchange Server 2010 - Security Considerations
- Microsoft Exchange Server 2010 - Limitations

Microsoft Exchange Server 2010 - Installation

Refer to the Microsoft Exchange Server 2010 documentation for installation information.

Microsoft Exchange Server 2010 - Configuration

To use Microsoft Exchange Server 2010 as the server for personal events, you must edit the Microsoft Exchange Server 2010 web service WSDL to specify the location of the web service.

To specify the location of the Microsoft Exchange Server 2010 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>
   ```

Microsoft Exchange Server 2010 - Security Considerations

Events includes a Microsoft Exchange Server 2010 adapter that communicates with the Microsoft Exchange Server 2010 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. You must enable Basic authentication. Further, you must enable anonymous access to `Services.wsdl`, `Messages.xsd`, and `Types.xsd` so that JAX-WS can access them to create the service port before committing any web service call. This involves creating a virtual directory and enabling anonymous authentication and disabling Windows authentication.

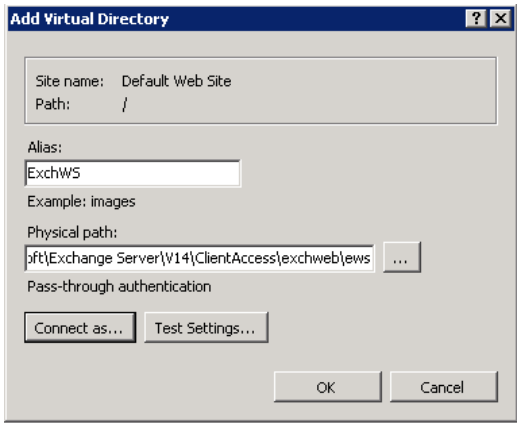
To edit Microsoft Exchange Server security settings:

1. On Microsoft Exchange Server, open Internet Information Services (IIS) Manager.
2. Under **Node *computer_name*** > **Sites >Default Web Site** > **EWS**, double-click **Authentication** under IIS.
3. Right-click **Basic Authentication** and select **Enable** to enable Basic Authentication for the EWS application.

4. Under Sites, right-click **Default Web Site** and select **Add Virtual Directory** to create a virtual directory that will be used to provide anonymous access to `Services.wsdl`, `Messages.xsd`, and `Types.xsd`.
5. In the Add Virtual Directory dialog, in the Alias field, specify the name of the virtual directory, for example `ExchWS`.
6. In the Physical Path field, specify the path to the virtual directory. For example (Figure 8-4):

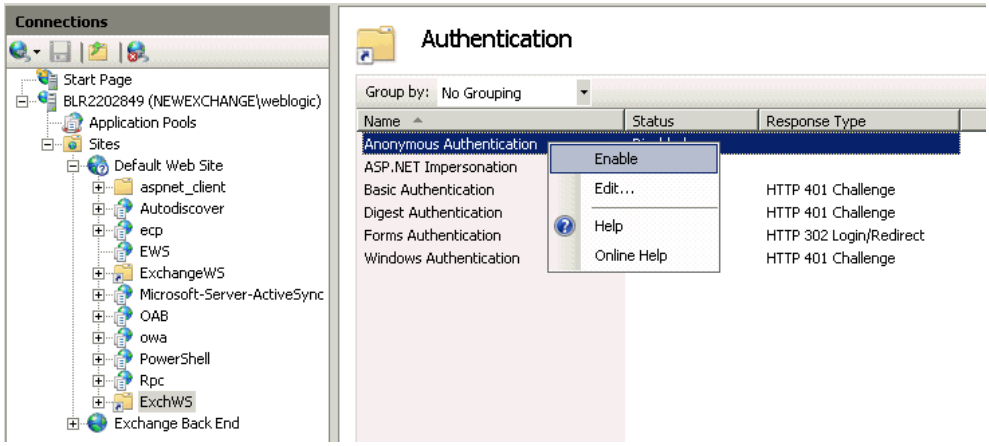
```
C:\Program Files\Microsoft\Exchange
Server\V14\ClientAccess\exchweb\ews
```

Figure 8-4 Creating a Virtual Directory



7. Click **Connect as**.
8. In the Connect As dialog, ensure **Application user (pass-thorough authentication)** is selected.
9. Click **OK**.
10. Under **Default Web Site > ExchWS**, double-click **Authentication** under IIS.
11. Right-click **Anonymous Authentication** and select **Enable** (Figure 8-5).

Figure 8-5 Enabling Anonymous Authentication



12. Right-click **Anonymous Authentication** and select **Edit**.
13. In the Edit Anonymous Authentication Credentials dialog, ensure **Application pool identity** is selected.
14. Right-click **Windows Authentication** and select **Disable**.

Events uses Basic Authentication to communicate with the Microsoft Exchange Server. To secure the communication, you must enable SSL. For more information, see:

`http://www.microsoft.com/technet/prodtechnol/WindowsServer2003/Library/IIS/56bdf977-14f8-4867-9c51-34c346d48b04.mspx?mfr=true`

You must also ensure that the `SSLAlwaysNegoClientCert` property is set to `true` in IIS. The `SSLAlwaysNegoClientCert` property controls SSL client connection negotiations.

For example, use the following command to set the `SSLAlwaysNegoClientCert` property:

```
CScript.exe adsutil.vbs SET w3svc/1/SSLAlwaysNegoClientCert true
```

For more information about the `SSLAlwaysNegoClientCert` property, see:

`http://www.microsoft.com/technet/prodtechnol/WindowsServer2003/Library/IIS/bce0fb87-79ea-40cd-963f-239545b61a12.mspx?mfr=true`

Microsoft Exchange Server 2010 - Limitations

There are currently no known limitations.

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 topics:

- Registering Events Servers Using Fusion Middleware Control
- Registering Event Servers Using WLST

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.
2. From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
3. On the WebCenter Portal Service Configuration page, select **Personal Events**.
4. To connect to a new events server instance, click **Add**.
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.

Table 8-2 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 2010** • **Microsoft Exchange Server 2013**
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 8-3 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.

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 createPersonalEventConnection and setPersonalEventConnection in *WebCenter WLST Command Reference Reference*.

> **Note:**
>
> To start using the new (active) connection you must restart the managed server on which WebCenter Portal is deployed.

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 topics:

- Choosing the Active Events Server Using Fusion Middleware Control
- Choosing the Active Events Server Connection Using WLST

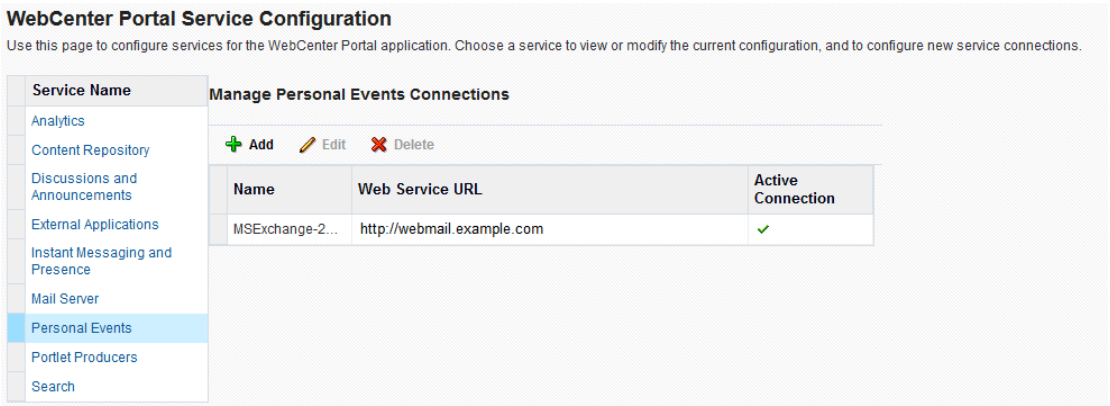
Choosing the Active Events Server Using Fusion Middleware Control

To change the active connection:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal.
2. From the **WebCenter Portal** menu, select **Settings** > **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 8-6 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.

Choosing the Active Events Server Connection Using WLST

Use the WLST command `setPersonalEventConnection` with `default=true` to activate an existing events server connection. For command syntax and examples, see setPersonalEventConnection in *WLST Command Reference for WebLogic Server*.

To subsequently disable an events connection, run the same WLST command with `default=false`. Connection details are retained but the connection is no longer named as an active connection.

> **Note:**
>
> To start using the active connection you must restart the managed server on which WebCenter Portal is deployed.

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:

- Modifying Events Server Connection Details Using Fusion Middleware Control
- Modifying Events Server Connection Details Using WLST

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.
2. From the **WebCenter Portal** menu, select **Settings** > **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 8-3
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.

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 setPersonalEventConnection in *WebCenter WLST Command Reference Reference* .

> **Note:**
>
> To start using the updated (active) connection you must restart the managed server on which WebCenter Portal is deployed.

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:

- Deleting Event Server Connections Using Fusion Middleware Control
- Deleting Event Server Connections Using WLST

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.
2. From the **WebCenter Portal** menu, select **Settings** > **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.

Deleting Event Server Connections Using WLST

Use the WLST command `deleteConnection` to remove an events server connection. For command syntax and examples, see deleteConnection in *WebCenter WLST Command Reference Reference*.

> **Note:**
>
> To effect this change you must restart the managed server on which WebCenter Portal is deployed.

9

Integrating Other Oracle Applications

Integrate other Oracle applications, such as Siebel, E-Business Suite, JD Edwards, PeopleSoft, and Oracle Business Intelligence, with WebCenter Portal.

> **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 granted through WebCenter Portal Administration. Aside from these permissions for WebCenter Portal, you may also need additional permissions for the other Oracle applications being integrated.
>
> For more information about roles and permissions, see Understanding Administrative Operations, Roles, and Tools.

Topics:

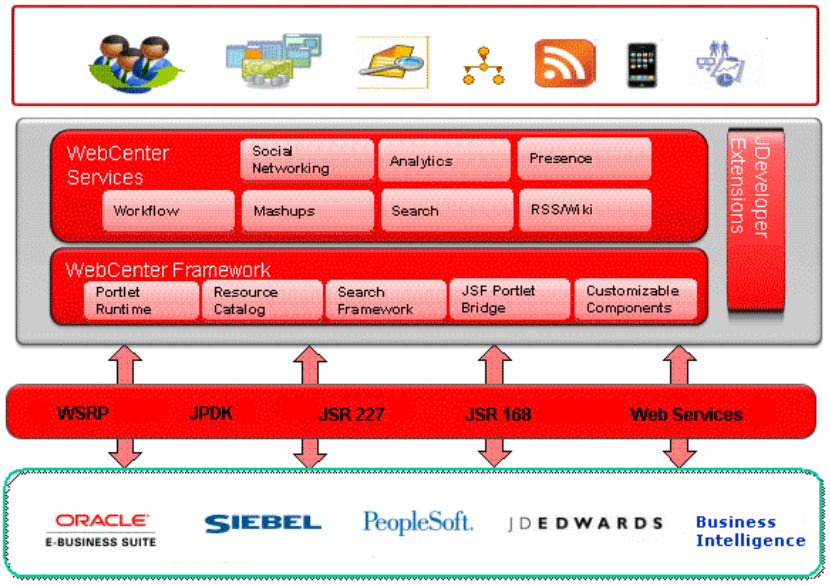
- About Integrating Other Oracle Applications
- Integrating Siebel Applications
- Integrating E-Business Suite Applications
- Integrating JD Edwards Applications
- Integrating PeopleSoft Applications
- Integrating Oracle Business Intelligence Presentation Services
- Integrating with Oracle Content and Experience Cloud

About Integrating Other Oracle Applications

Oracle WebCenter Portal is an integrated suite of technology designed to deliver a unified, context-aware user experience. WebCenter Portal integrates structured and unstructured content, business intelligence, business processes, communication, and collaboration services, and removes the boundaries between enterprise applications. By integrating other applications available within the enterprise with WebCenter Portal, you can create context-centric, composite applications that leverage the capabilities of these applications, extending WebCenter Portal and changing the way people work.

WebCenter Portal uses industry-standard technologies to integrate (primarily as WSRP portlets) other application components. Figure 9-1 shows the technologies involved in WebCenter Portal integration with other Oracle applications.

Figure 9-1 WebCenter Portal Integration



Although not all applications support the same integration mechanisms, the integration process is generally quite simple, consisting of exposing the application object to be integrated as a portlet, registering the portlet with WebCenter Portal, adding the portlet to a page, and then running and testing the results.

In Figure 9-1 we show the applications that can be integrated as Siebel, E-Business Suite, JD Edwards, PeopleSoft, and Oracle Business Intelligence. These Oracle applications are fully supported and documented within this chapter. However, you can integrate virtually any application that can expose objects as WSRP portlets. The process for integrating them is the same as for the Oracle applications documented here: expose the object as a portlet, register the portlet in WebCenter Portal, and add the portlet to a page. Refer to the documentation for one of the supported Oracle applications for a description of how to consume an exposed portlet in WebCenter Portal.

Integrating Siebel Applications

This section describes how to integrate a Siebel Web service in WebCenter Portal. Siebel and WebCenter can work together to include Siebel's CRM capabilities as portlets within WebCenter Portal.

- How to Integrate Siebel Applications as Web Services

How to Integrate Siebel Applications as Web Services

This section describes how to integrate Siebel applications as Web services in WebCenter Portal.

To be able to add a Siebel Web service data control or a task flow containing a data control to a portal page you must first have configured WS-Security for WebCenter Portal. For more information about configuring WS-Security, see Configuring Web Services Security login credentials in Oracle Fusion Middleware Administering Oracle WebCenter Portal. For more information about Web service data controls, see Creating Data Controls and Web Service Data Controls in Oracle Fusion Middleware Building Portals with Oracle WebCenter Portal.

This section contains the following subsections:

- How to Prepare the Siebel Application
- How to Consume a Siebel Web Service Data Control

How to Prepare the Siebel Application

This section describes how to create an inbound Web service, set up operations for the inbound service, and generate a WSDL that you will later use to create a data control in WebCenter Portal.

This section contains the following subsections:

- How to Create an Inbound Web Service
- How to Create Operations for the Inbound Web Service

How to Create an Inbound Web Service

To create an inbound Web service:

1. Log into the Siebel application as an administrator.
2. Navigate to the Administration - Web Services page.
3. Click **Inbound Web Services**.

   The Inbound Web Services page shows the out-of-the-box Web services and any other Web services that are currently exposed.
4. Click **Menu** and select `New Record` from the drop-down list.
5. Enter the values for **Namespace**, **Name**, **Status** and **Comment** as appropriate for the Web service you want to set up. For example:

Field Name	Value
Namespace	`http://xmlns.oracle.com`
Name	`Siebel Customer Account`
Status	`Active`
Comment	`For Fusion Middleware`

6. Scroll to the Service Ports pane and select **New Record** from the **Menu** drop-down list.

7. Enter `CustAccount` as the **Name** and click **Type**.
8. In the **Inbound Web Service Port Type** pick applet, open the New tab.
9. Select `Business Service` as the **Implementation Type**.
10. From the **Service Name** list, select `Siebel Account`.
11. In the **Inbound Web Service Port Type** pick applet, click **OK** to create the inbound Web service.
12. From the Service Ports dialog's **Transport** drop-down list, select `HTTP`.
13. In the **Address** field, set the URL to your Siebel instance. For example:

    ```
    http://xmlns.oracle.com/eai_enu/start.swe?
    SWEExtSource=WebService&SWEExtCmd=Execute&UserName=SADMIN&Password=SADMIN
    ```

14. From the **Menu** drop-down list, select `Save Record`.

How to Create Operations for the Inbound Web Service

After creating the inbound Web service, continue by adding operations to the inbound Web service and then create a WSDL file, follow these steps:

1. Scroll to the Operations section and select `New` from the **Menu** drop-down list.
2. In the **Operation Name** field, enter `AccountInsert`.
3. Click **Method Display Name** to open the Business Service Method dialog.
4. Select `Insert` as the **Method**, and click **OK**.
5. From the **Authentication Type** drop-down list, select an appropriate authentication type:

Authentication Type	Session Type	Description
None	None	A single request is sent with an anonymous user login, and the session is closed after the response is sent out. In order for the anonymous session to be identified by the SWSE Plug-in, UsernameToken and PasswordText must be excluded in the SOAP headers.
Username and password	None	A single request is sent with the username and password used to log in, and the session is closed after the response is sent out.
Username and password	Stateless	The initial request to log in establishes a session that is to remain open and available for subsequent requests. Username/password are used to log in and a session token is returned in a SOAP header included in the outbound response. The session remains open.
Session token (stateless)	Stateless	Request to reconnect to an established session, using the information contained in the session token. If the session has been closed, automatic re-login occurs. The Siebel servers include the session token in the SOAP header of the response. The session remains open.

Authentication Type	Session Type	Description
Session token (stateless)	None	When a SOAP header carries a session token and has the session type set to None, then the Session Manager on the SWSE closes (logs out) of this session, and invalidates the session token. The session token is not used after the session is invalidated.

6. Click **New** to create a new operation.
7. In the **Operation Name** field, enter a name for the new operation (for example, `AccountQueryByExample`).
8. Click **Method Display Name** for the new operation.
9. In the Business Service Method dialog, select the query method (for example, **Query By Example**) and click **OK**.
10. Continue by adding any additional operations you may need as described in steps 6 to 9 above.
11. In the Service Ports pane, select `Save Record` from the **Menu** drop-down list.
12. In the Inbound Web Services pane, select `Save Record` from the **Menu** drop-down list.
13. Select `Clear Cache` from the **Menu** drop-down list.
14. Click the **Generate WSDL**.
15. On the File Download dialog, click **Open**.
16. Select **File --> Save As...**
17. Locate the directory where you want to save the WSDL file, enter a name for the file and click **Save**.

How to Consume a Siebel Web Service Data Control

This section describes how you can create a Web Service data control and add it to a portal page. The steps in this section assume that you have prepared the application and generated a WSDL as described in How to Prepare the Siebel Application.

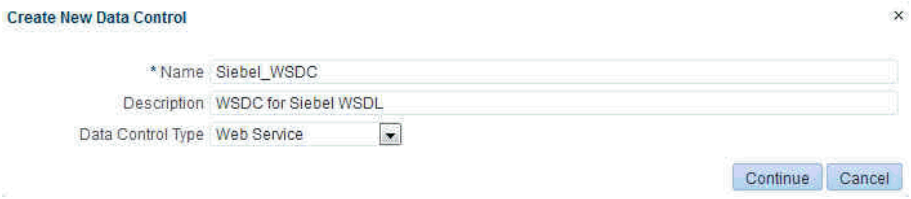
> **Note:**
>
> Before you can add a data control or task flow containing a data control to a portal page you must first have configured WS-Security for WebCenter Portal. For more information about configuring WS-Security, see Configuring Web Services Security login credentials in *Administering Oracle WebCenter Portal*.

To create a Web service data control:

1. In WebCenter Portal or the portal in which you want to create the data control, go to either the **Shared Assets** or **Assets** page.
2. Select **Data Controls** and click **Create**.

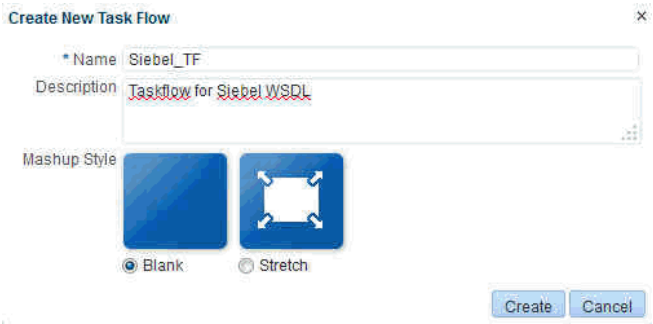
   The Create New Data Control dialog displays.

Figure 9-2 Create New Data Control Dialog



3. In the Create New Data Control dialog, enter a **Name** and **Description** for the data control, select `Web Service` as the **Data Control Type,** and then click **Continue**.
4. Enter the WSDL URL that you generated earlier and other details for the data control and click **Continue**.
5. Click **Show Methods**.
6. Select the method(s) to make available and click **Next**.
7. Enter the parameter default values, if any, and click **Create**.
8. To make the data control available, from the **Shared Assets** or **Assets** page, select **Task Flows**. The Create New Task Flow dialog displays.

Figure 9-3 Create New Task Flow Dialog



9. Enter the task flow **Name** and **Description**, select the **Mashup Style** to use, and then click **Create** to create the task flow.
10. Select the task flow and click the **Edit** icon.
11. Add the data control (with parameter form) as a table onto the task flow and verify the data.
12. To make the task flow available, navigate to **Administration > Business Role Pages**.
13. Select **Business Role Page** and click the **Create** icon.
14. Edit the page and save the changes.
15. Drop the task flow onto the page and verify the data.

Integrating E-Business Suite Applications

This section describes how to integrate E-Business Suite applications in WebCenter Portal.

This section contains the following subsections:

- About Integrating EBS Applications
- Required Configurations for Integrating EBS
- How to Integrate EBS Applications as WSRP Portlets
- How to Integrate EBS Applications as Data Controls

About Integrating EBS Applications

This section describes the integration points and requirements integrating Oracle E-Business Suite portlets in WebCenter Portal.

This section includes the following subsections:

- Understanding EBS Integration
- Requirements for Integrating EBS Applications

Understanding EBS Integration

Out of the box, Oracle E-Business Suite OA Framework-based portlets, such as Applications Navigator, and Favorites are WSRP and JSR 168-compliant. That means that you can access these Oracle E-Business Suite portlets from WSRP-compliant portal servers, such as WebCenter Portal, by simply adding the portlet onto a page. Follow the instructions in How to Add the EBS Portlet to a Portal Page to add them to a WebCenter Portal or portal page.

You can also create new E-Business Suite portlets that are WSRP and JSR 168-compliant that can similarly be added to WebCenter Portal. Creating and consuming WSRP and JSR 168 compliant portals in WebCenter Portal is described in How to Integrate EBS Applications.

Requirements for Integrating EBS Applications

The following requirements apply for integrating Oracle E-Business Suite portals in WebCenter Portal:

- Regions to be exposed as portlets must be created using Oracle E-Business Suite OA Framework Release 12 as previous versions are not WSRP/JSR 168-compliant.
- Oracle E-Business Suite can be configured to use Oracle Internet Directory (OID) and one of following single sign-on solutions:

> **Caution:**
>
> Both WebCenter Portal and Oracle E-Business Suite must share the same OID instance and user IDs.

  - Oracle Single Sign-On (OSSO)
  - Oracle Access Manager (OAM)

  If you are using OSSO, follow the steps in My Oracle Support document 376811.1 to integrate E-Business Suite Release 12 with OID and OSSO.

If you are using OAM, follow the steps in My Oracle Support document 975182.1 to integrate E-Business Suite Release 12 with OAM.

E-Business Suite can also be configured to OID without OAM or OSSO. For more information, see How to Prepare OID for Use Without Single Sign-On.

> **Note:**
>
> Although Oracle E-Business Suite can be configured to use Oracle Internet Directory (OID) without single sign-on, this is not a recommended approach as users will be prompted for credentials each time they move to or from the integrated portal or data control.

- You must have granted WebCenter Portal access to the E-Business Suite Portlet Producer and added and configured the appropriate users.

> **Note:**
>
> To complete some steps, you may need system administrator permissions.

Required Configurations for Integrating EBS

This section contains configurations that should be undertaken prior to attempting to integrate portal or data controls in WebCenter Portal.

This section contains the following subsections:

- How to Prepare OID for Use Without Single Sign-On
- How to Create a User in EBS and Assign a Responsibility
- How to Configure the EBS Applications Profile Options
- How to Add the WebCenter Host as a Trusted Portal Using AutoConfig

How to Prepare OID for Use Without Single Sign-On

This section describes the steps to configure OID as an optional standalone environment without using either OAM or OSSO. Note that this is not a recommended approach as users will be prompted for credentials each time they move to or from an integrated portal or data control. If you have installed an SSO solution, continue with How to Create a User in EBS and Assign a Responsibility.

> **Caution:**
>
> Both WebCenter Portal and Oracle E-Business Suite must share the same OID instance and the same user IDs.

1. Register the OID instance on the host server by following the steps below:

a. Run the following command:

```
$FND_TOP/bin/txkrun.pl -script=SetSSOReg -registerinstance=yes
```

b. Supply the required information at the following prompts:

```
Enter the host name where the Oracle iAS Infrastructure database is
installed ? <Enter the OID Host>
Enter the LDAP Port of the Oracle Internet Directory server ? <Enter the LDAP
Port>
Enter SSL LDAP Port of the Oracle Internet Directory server ? <Enter the LDAP
SSL Port>
Enter the Oracle Internet Directory Administrator (orcladmin) Bind password ?
<Password>
Enter Oracle E-Business apps database user password ? <Password>
```

c. Restart all the services by navigating to `$ADMIN_SCRIPTS_HOME` and running:

```
./adstpall.sh apps/<apps to stop>
```

and then:

```
./adstrtal.sh apps/<apps to start>
```

2. Register OID by following the steps below:

a. Run the following command:

```
$FND_TOP/bin/txkrun.pl -script=SetSSOReg -registeroid=yes
```

b. Supply the required information at the following prompts:

```
Enter LDAP Host name ? <Enter the OID Host>
Enter the LDAP Port on Oracle Internet Directory server ? <Enter the LDAP Port>
Enter the Oracle Internet Directory Administrator (orcladmin) Bind password ?
<Password>
Enter the instance password that you would like to register this application
instance with ? <Password>
Enter Oracle E-Business apps database user password ? <Password>
```

c. Restart all the services by navigating to `$ADMIN_SCRIPTS_HOME` and running:

```
./adstpall.sh apps/<apps to stop>
```

and then:

```
./adstrtal.sh apps/<apps to start>
```

How to Create a User in EBS and Assign a Responsibility

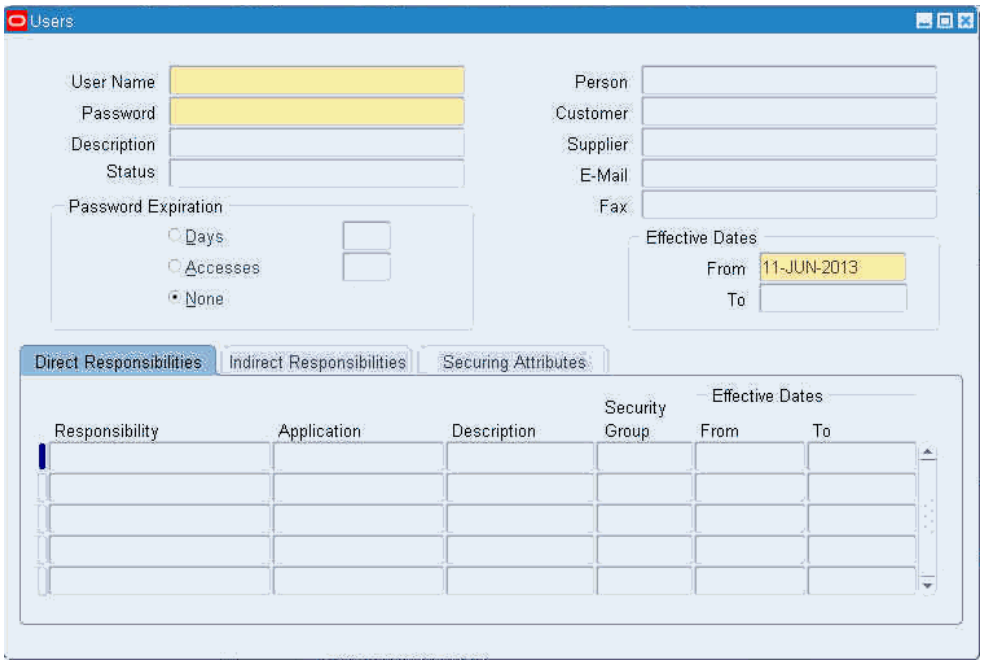
For integration with EBS to work, WebCenter Portal and EBS must have a common OID identity store. With a common OID, you can either create a new user in EBS, or use an existing user in OID, and then assign a responsibility to that user. This will ensure that the user has access to the portlets in WebCenter Portal.

To create a new user and assign a responsibility:

1. Log into EBS as a system administrator if not already logged in.
2. In the Navigation pane, expand the System Administrator node, expand Security, expand User, and then click **Define**.

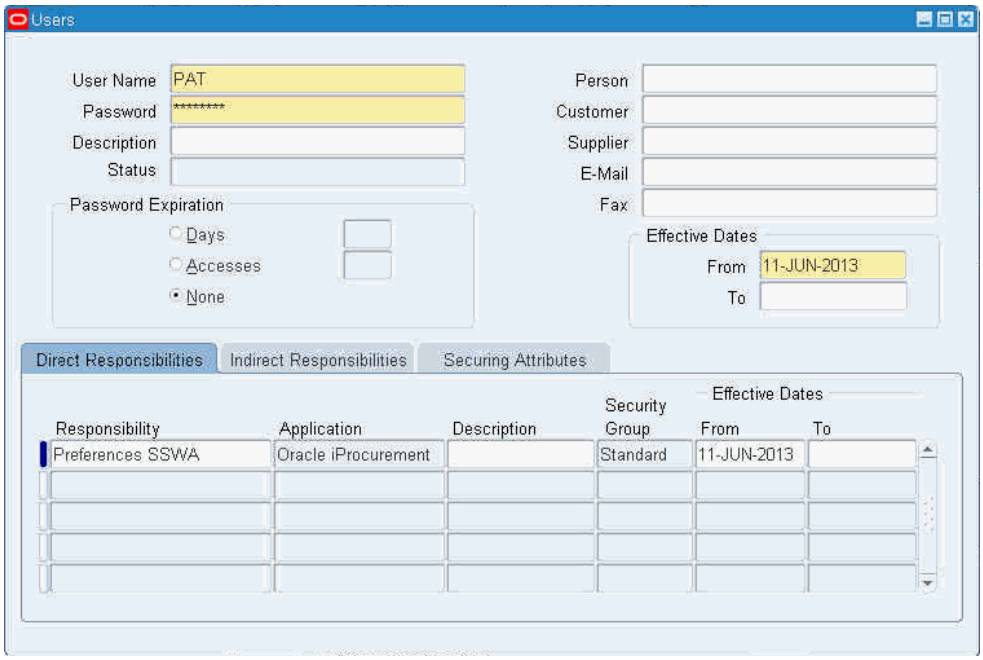
   The Users window displays.

Figure 9-4 Users Window



3. Enter the **User Name**, and **Password**. The Password Expiration options should be set to **None**.
4. Open the Direct Responsibilities tab, and search for the **Responsibility** to add and assign the **Application** to associate with it (for example, search for `Preferences SSWA` and assign `Oracle iProcurement` to it), and then click **Save**.

Figure 9-5 Users Window Showing the Direct Responsibilities Tab



5. To confirm, log in with the newly created user and check that the application associated with the Responsibility is listed.

How to Configure the EBS Applications Profile Options

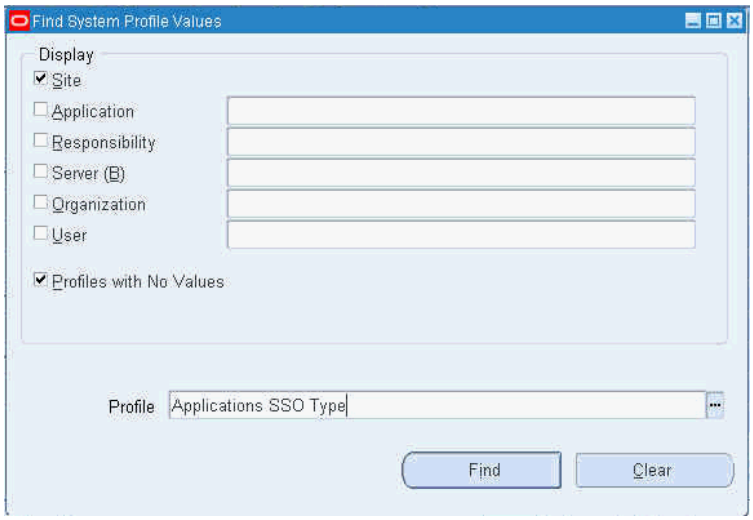
This section describes how to configure EBS Applications Profile Options and is a requirement for both SSO and non-SSO configurations.

To configure the EBS profile options:

1. Log into EBS as a system administrator.
2. In the Navigation pane, expand the System Administrator node, and then click **Define Profile Options**.
3. Close the Profiles window.
4. In the Navigator, select **Profile System Values** and click **Open**.

   The Find System Profile Values window displays.

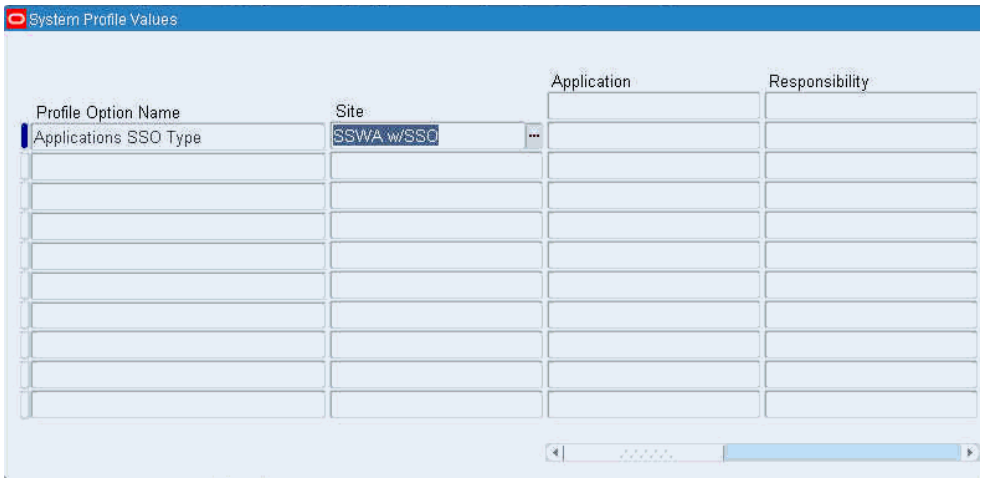
Figure 9-6 Find System Profile Values Window



5. Enter the **Profile** name to update and click **Find**.

   The System Profile Values window displays.

Figure 9-7 System Profile Values Window



Update the values for the following profiles, saving your entries after each update:

`Applications SSO Enable OID Identity Add Event =`**`Enabled`**

Applications SSO Login Types =**`Both`**

Application SSO LDAP Synchronization =Enabled

Applications SSO Type =**`SSWA w/ SSO`**

Link Applications user with OID user with same username =Enabled

6. Restart all the services by navigating to `$ADMIN_SCRIPTS_HOME` and running:

   `./adstpall.sh apps/<apps to stop>`

   and then:

   `./adstrtal.sh apps/<apps to start>`

How to Add the WebCenter Host as a Trusted Portal Using AutoConfig

The EBS WSDL is protected and before you can access it you must first add an entry for the consuming WebCenter Portal instance's host using the EBS AutoConfig tool. Note that without this configuration step you will get a "403 Forbidden" error if you try to access the WSDL.

To add the WebCenter Portal host as a trusted portal:

1. Log into EBS as a system administrator if you are not already logged in.
2. In the Navigation pane, expand the System Administrator node, expand Oracle Applications Manager, and then click **Workflow**.
3. Open the Sitemap tab and click **AutoConfig**.
4. In the Edit Parameter column, click the **Edit** icon in the Applications Tier row.
5. Open the System tab and expand the `oa_web_server` node.
6. In the list of nodes, look for any that have access to Portlet Producer URLs, add the WebCenter Host and click **Save**. If you need to add multiple host name, add them separated by space.

7. Run the autoconfig script entering `apps` as the password when prompted:

   ```
   cd $ADMIN_SCRIPTS_HOME
   ./adautocfg.sh
   ```

8. Restart all the services by running the following commands from `$ADMIN_SCRIPTS_HOME`:

   ```
   ./adstpall.sh apps/<apps to stop>
   ```

   and then:

   ```
   ./adstrtal.sh apps/<apps to start>
   ```

How to Integrate EBS Applications as WSRP Portlets

This section describes how to integrate EBS regions as WSRP portlets in WebCenter Portal. To start, you'll need to generate the portlet for the region using the Portlet Generator, and then continue by registering the producer and integrating it in WebCenter Portal.

This section contains the following subsections:

- How to Prepare the EBS Portlet for Remote Access
- How to Integrate EBS Applications

How to Prepare the EBS Portlet for Remote Access

Oracle E-Business Suite provides a tool called Portlet Generator to convert existing standalone Oracle Application Framework regions into portlets. To be available for portletization, a region must have the following properties.

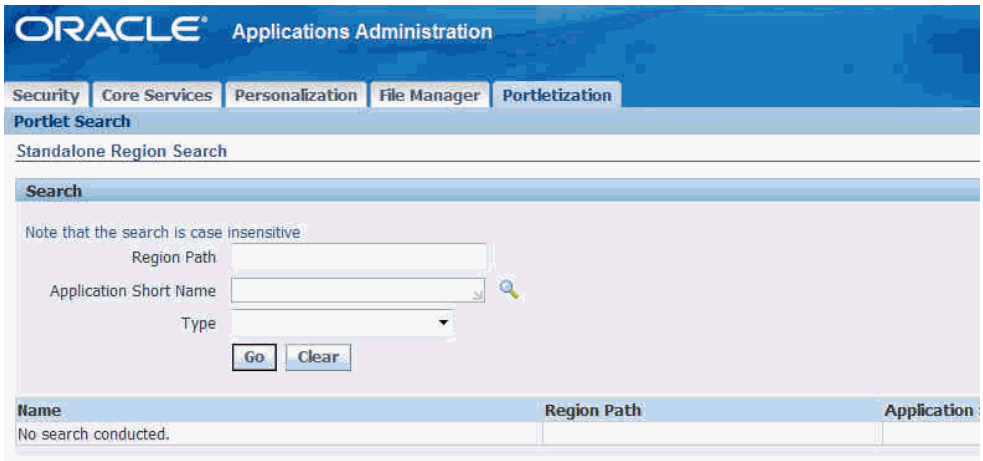
- Regions must have an Application Module (AM) defined and must have its standalone property set to `true`.
- Inline regions must have an AM defined and have its standalone property set to `true`.
- Content regions must have an AM defined (content regions do not have a standalone property)

To expose EBS functionality as a portlet using Portlet Generator:

1. Log into EBS as a system administrator.
2. In the Navigation pane, expand the Functional Administrator node, and then click **Home**.

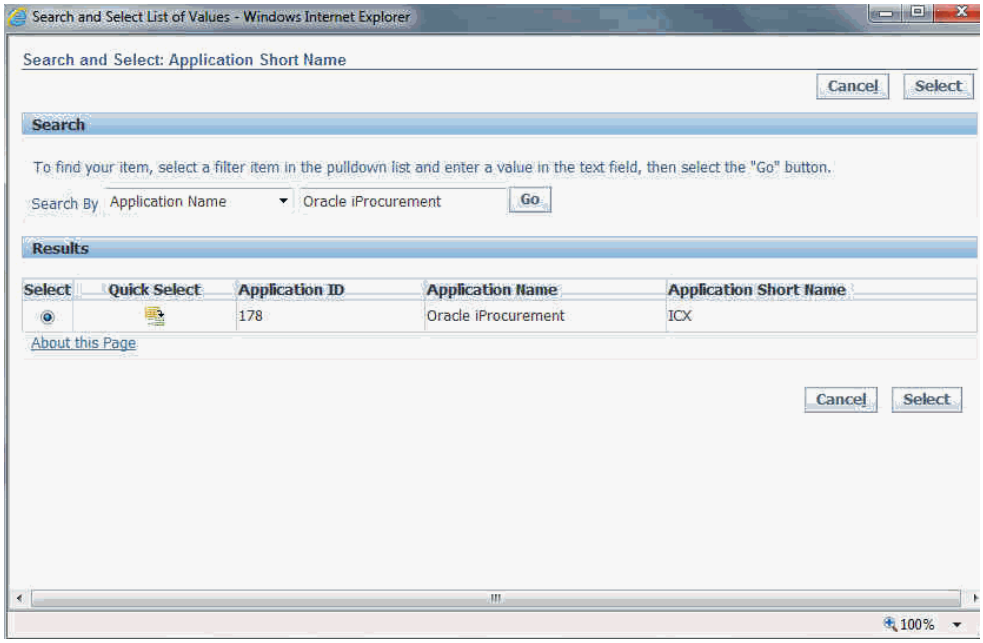
   The Application Administration page displays (see Figure 9-8).

Figure 9-8 Application Administration Page



3. Open the Portletization tab and click the **Search** icon for the **Application Short Name** field (or enter the **Application Short Name** if you know it).

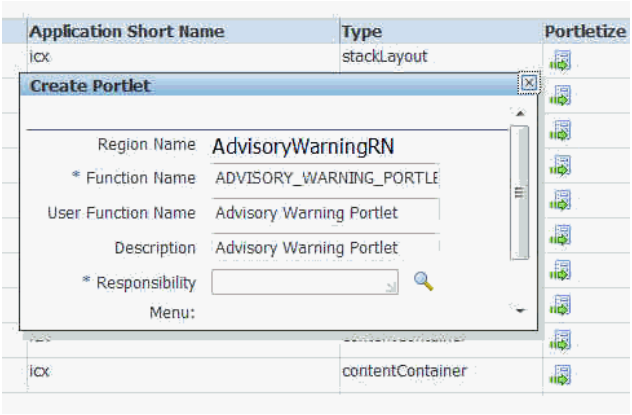
Figure 9-9 Application Short Name Search Dialog



4. Select the Search By criteria (for example, select `Application Name` and enter `Oracle iProcurement`) and search for the functionality to portletize.
5. Select the row returned in the search results and click **Select**.
6. Click **Go** to list the EBS functionality that can be portletized.
7. Click the **Portletize** icon for the functionality you want to expose (for example, `AdvisoryWarningRN`).

   The Create Portlet dialog displays (see Figure 9-10).

Figure 9-10 Create Portlet Dialog



8. Enter the **Responsibility** to associate the region with (for example, `Preferences SSWA`) or use the **Search** function.
9. Click **Apply**.
10. Continue by registering the EBS producer and integrating it in a portal (see How to Integrate EBS Applications).

How to Integrate EBS Applications

This section contains the following subsections:

- How to Prepare the EBS Portlet for Remote Access
- How to Register the EBS WSRP Producer in WebCenter Portal
- How to Add the EBS Portlet to a Portal Page
- How to Test the Portlet Connection

How to Prepare the EBS Portlet for Remote Access

Prepare the standalone regions to be portletized as described in the section on How to Prepare the EBS Portlet for Remote Access.

Before adding the portlets in WebCenter Portal, be sure to bounce the Apache listener as the menu and function definitions are cached.

How to Register the EBS WSRP Producer in WebCenter Portal

You can register the EBS WSRP producer directly in WebCenter Portal using Fusion Middleware Control.

To register the EBS WSRP producer using Fusion Middleware Control:

1. Prepare the EBS page that you want to consume in WebCenter Portal for remote access as described in How to Prepare the EBS Portlet for Remote Access.
2. Log in to Fusion Middleware Control for the WebCenter Portal domain (`WC_Domain` by default).
3. Expand `WebCenter Portal` in the Navigation bar and from the WebCenter Portal menu, and select **Register Producer**.

The Add Portlet Producer page displays.

4. Enter a **Connection Name**, set the **Producer Type** to `WSRP Producer`, and paste the WSDL endpoint URL that you copied in step 1 into the **URL End Point** field.
5. Click **OK** and verify that the producer connection was created successfully.
6. Continue by adding the portlet to a portal page as described in How to Add the EBS Portlet to a Portal Page.

How to Add the EBS Portlet to a Portal Page

Follow the steps below to consume the EBS remote producer in a WebCenter Portal page:

1. Log into WebCenter Portal and, optionally, open the target portal.
2. Go to the page, or create a new page, where you want to add the EBS portal.
3. Click **Add Content** and in the resource catalog, select **UI Components** and then **Portlets**.

   Note that if you've created a custom resource catalog, **Portlets** may not appear. In this case, you will need to add it to the resource catalog. For information about managing resource catalogs, see Working with Resource Catalogs in *Building Portals with Oracle WebCenter Portal*.
4. Click the portlet you added in Fusion Middleware Control.
5. Click **Add** for the EBS portlet you want to add to your portal page.
6. On the portal page, click the portlet's **View Actions** menu, and select **Display Options**.
7. In the Display Options dialog, set **Render Portlet in IFrame**to `True` and click **OK**.
8. Continue by checking the portlet connection as described in How to Test the Portlet Connection.

How to Test the Portlet Connection

Follow these steps to test the portlet connection by modifying content and checking that the modification shows up in the EBS application.

1. On the WebCenter Portal or portal page to which you added the EBS portlet, modify some information that you can verify the changes for in the EBS application.
2. Save your changes and confirm that the changes also appear in the EBS application.

How to Integrate EBS Applications as Data Controls

This section describes how to add EBS applications as Web service data controls on a WebCenter Portal page.

This section contains the following topics:

- How to Generate the WSDL
- How to Add a Web Service Data Control to a Portal Page

How to Generate the WSDL

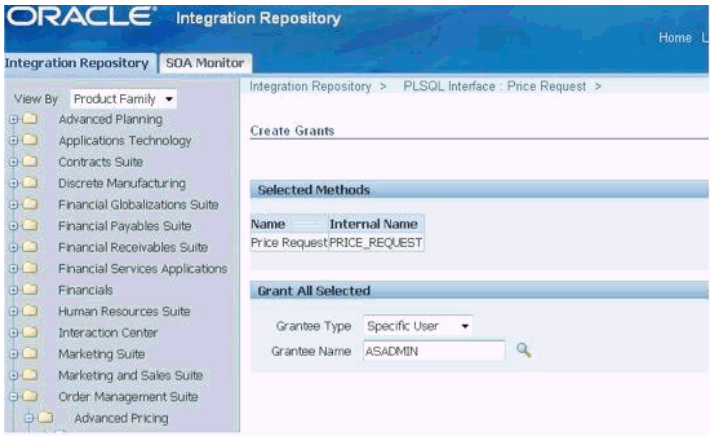
This section describes how to create the WSDL.

1. Log into E-Business Suite as the SYSADMIN user.
2. In the Navigation pane, expand the Integrated SOA Gateway node and under the Integrated SOA Gateway sub-node click **Integration Repository.**
3. From the Integration Repository tab, navigate to the part of the EBS application to expose. For example, for the price request interface, you would go to **Order Management Suite > Advanced Pricing > Price List**, and then selecting **Price Request** from the list of integration points.
4. Click **Generate WSDL** to expose the integration point (for our example, a PL/SQL API integration point) as a Web service.
5. Right-click the **View WSDL** link and open the link in a new tab or new window (be sure to keep the tab or window open as you'll need it later).
6. On the Integration Repository page under Procedures and Functions (see Figure 9-11), check the box for the object to grant access to, and then click **Grant Access**.

Figure 9-11 Integration Repository - Price Request Example

7. Select the **Grantee Type** and **Grantee Name** (the user you want to grant access to the exposed object), or use the Search tool. For our example, we will grant access to ASADMIN.

Figure 9-12 Integration Repository - Create Grants Page



How to Add a Web Service Data Control to a Portal Page

Once you have the WSDL, you can continue by using it to create a web service data control.

> **Note:**
>
> Before you can add a data control or task flow containing a data control to a portal page you must first have configured WS-security for WebCenter Portal. For more information about configuring WS-security, see Configuring Web Services Security.

For more information about creating a web service data control, see Creating a Web Service Data Control in *Building Portals with Oracle WebCenter Portal*. For information about web service data controls, see About Web Services Data Controls in *Building Portals with Oracle WebCenter Portal*.

To create a web service data control:

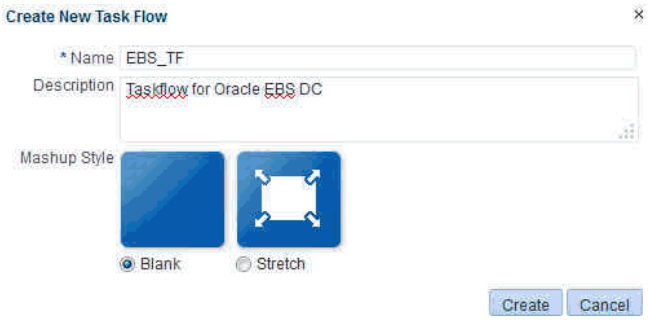
1. In WebCenter Portal or the portal in which you want to create the data control, go to either the **Shared Assets** or **Assets** page.
2. Select **Data Controls** and click **Create**.

   The Create New Data Control dialog displays (see Figure 9-31).

Figure 9-13 Create New Data Control Dialog

3. In the Create New Data Control dialog, enter a **Name** and **Description** for the data control, select `Web Service` as the **Data Control Type,** and then click **Continue**.
4. Enter the WSDL URL that you generated in How to Generate the WSDL and other details for the data control and click **Continue**.
5. Click **Show Methods**.
6. Select the method(s) to make available and click **Next**.
7. Enter the parameter default values, if any, and click **Create**.
8. To make the data control available, from the **Shared Assets** or **Assets** page, select **Task Flows**. The Create New Task Flow dialog displays (see Figure 9-33).

Figure 9-14 Create New Task Flow Dialog



9. Enter the task flow **Name** and **Description**, select the **Mashup Style** to use click **Create** to create the task flow.
10. Select the task flow and click the **Edit** icon.
11. Add the data control (with parameter form) as a table onto the task flow and verify the data.
12. To make the task flow available, navigate to **Administration > Business Role Pages**.
13. Select **Business Role Page** and click the **Create** icon.
14. Edit the page and save the changes.
15. Drop the task flow onto the page and verify the data.

Integrating JD Edwards Applications

This section describes how to integrate JD Edwards applications into WebCenter Portal.

This section contains the following subsections:

- How to Prepare the JD Edwards Application for Remote Access
- How to Register the Producer
- How to Add the JD Edwards Portlet to a WebCenter Portal Page
- How to Test the Portlet Connection

How to Prepare the JD Edwards Application for Remote Access

Before you can add JD Edwards standalone regions to WebCenter Portal, you must first prepare them to be portletized within JD Edwards by making them available externally as

portlets and locating the pre-configured WSDL in the `webclient.war/wsdl` directory. The WSDL URL is needed so that you can register the JD Edwards WSRP producer and consume it from a WebCenter Portal or portal page. To view the XML content of the JDE WSDL in the browser, open the Page source of the page in the browser. For more information, see Administering WSRP with Oracle WebCenter.

How to Register the Producer

You can register the JD Edwards WSRP producer directly in WebCenter Portal, as described in the Registering Portlet Producers in *Administering Oracle WebCenter Portal*. You can also register the JD Edwards WSRP producer using Fusion Middleware Control as described in the steps below.

To register the JD Edwards WSRP producer using Fusion Middleware Control:

1. Prepare the JD Edwards page that you want to consume in WebCenter Portal for remote access.
2. Log into Fusion Middleware Control for the WebCenter Portal domain (`WC_Domain` by default).
3. Expand `WebCenter Portal` in the Navigation bar and from the WebCenter Portal menu, and select **Register Producer**.

   The Add Portlet Producer page displays.
4. Enter `JDE` as the **Connection Name**, set the **Producer Type** to `WSRP Producer`, and paste the WSDL endpoint URL that you copied in step 1 into the **URL End Point** field.
5. Click **OK** and verify that the producer connection was created successfully.
6. Continue by adding the portlet to a portal page as described in How to Add the JD Edwards Portlet to a WebCenter Portal Page.

How to Add the JD Edwards Portlet to a WebCenter Portal Page

Follow the steps below to consume the JD Edwards remote producer in WebCenter Portal:

1. Log into WebCenter Portal.
2. Go to the page, or create a new page, where you want to add the JD Edwards portal.
3. Click **Add Content** and in the resource catalog, select **UI Components** and then **Portlets**.

   Note that if you've created a custom catalog, **Portlets** may not appear. In this case, you will need to add it to the resource catalog. For information about managing resource catalogs, see Working with Resource Catalogs in *Building Portals with Oracle WebCenter Portal*.
4. Click the portlet you added in Fusion Middleware Control.
5. Click **Add** for the JD Edwards portlet you want to add to your portal page.
6. On the portal page, click the portlet's**View Actions** menu, and select **Display Options**.
7. In the Display Options dialog, set**Render Portlet in IFrame** to `True` and click **OK**.

8. Continue by checking the portlet connection as described in How to Test the Portlet Connection.

How to Test the Portlet Connection

Follow these steps to test the portlet connection by modifying content and checking that the modification shows up in the JD Edwards application.

1. On the WebCenter Portal or portal page that you added the JD Edwards portlet to, modify some information that you can verify the changes for in the JD Edwards application.
2. Save your changes and confirm that the changes also appear in the JD Edwards application.

Integrating PeopleSoft Applications

This section describes how to integrate PeopleSoft applications in WebCenter Portal.

This section contains the following subsections:

- About Integrating PeopleSoft Applications
- How to Integrate PeopleSoft Applications as WSRP Portlets
- How to Integrate PeopleSoft Applications as Data Controls in WebCenter Portal

About Integrating PeopleSoft Applications

This section describes the benefits and methods involved in integrating PeopleSoft applications in WebCenter Portal.

This section includes the following subsections:

- Understanding PeopleSoft Integration
- Requirements for Integrating PeopleSoft Applications

Understanding PeopleSoft Integration

PeopleTools 8.51 and later lets you expose PeopleSoft applications as WSRP portlets in remote applications such as WebCenter Portal. This allows people who only need access to a small portion of PeopleSoft's functionality to access it through WebCenter Portal without needing to open or learn the entire PeopleSoft application.

Requirements for Integrating PeopleSoft Applications

This section the prerequisites for integrating PeopleSoft objects in WebCenter Portal.

- PeopleSoft 9.0 or later.
- PeopleTools 8.51 or later.
- When using WS-Security for automatic sign on to PeopleSoft, in order for the SAML assertion to be valid, the date/time on the PeopleSoft and Oracle WebCenter Portal servers must be synchronized. If this is problematic, then the PeopleSoft web server's time may be set to be slightly ahead of the Oracle WebCenter Portal server.

- For PeopleTools 8.51, you may need to create and configure a custom OWSM policy in order to fully support WS-Security. For more information, see How to Configure WS-Security for PeopleTools 8.51.
- For PeopleTools 8.51, only upper case subject names are supported, requiring that only fully upper case user IDs can be used in WebCenter for the integration to work.

How to Integrate PeopleSoft Applications as WSRP Portlets

This section describes how to expose PeopleSoft applications as WSRP portlets in WebCenter Portal.

This section includes the following subsections:

- How to Prepare the PeopleSoft Application for Remote Access
- How to Configure WS-Security for PeopleTools 8.52 and Later
- How to Attach a WS-Security Policy to WebCenter Portal
- How to Integrate PeopleSoft Applications in WebCenter Portal
- How to Configure WS-Security for PeopleTools 8.51

How to Prepare the PeopleSoft Application for Remote Access

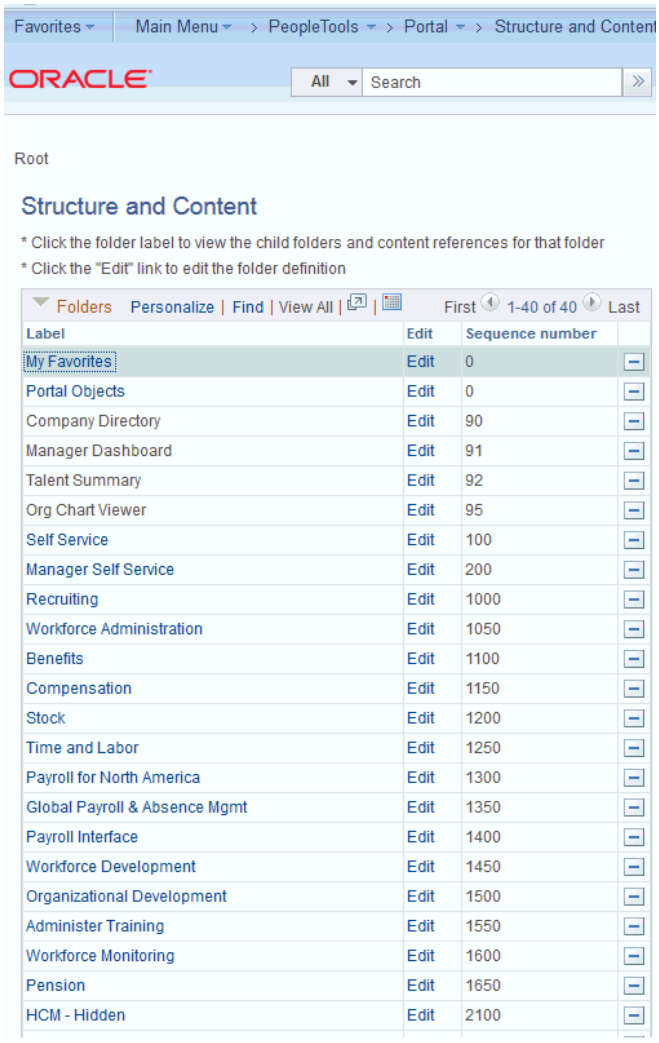
This section describes how to prepare the PeopleSoft application so that it can be consumed by WebCenter Portal.

To prepare the PeopleSoft application:

1. Log into PeopleSoft as an administrator.
2. Select PeopleTools from the main menu.
3. From the People Tools main menu, expand **Portal**.
4. Select **Structure and Content**.

   The Structure and Content page displays a list of folders containing PeopleSoft objects that could be exposed as a WSRP Portlet (see Figure 9-15).

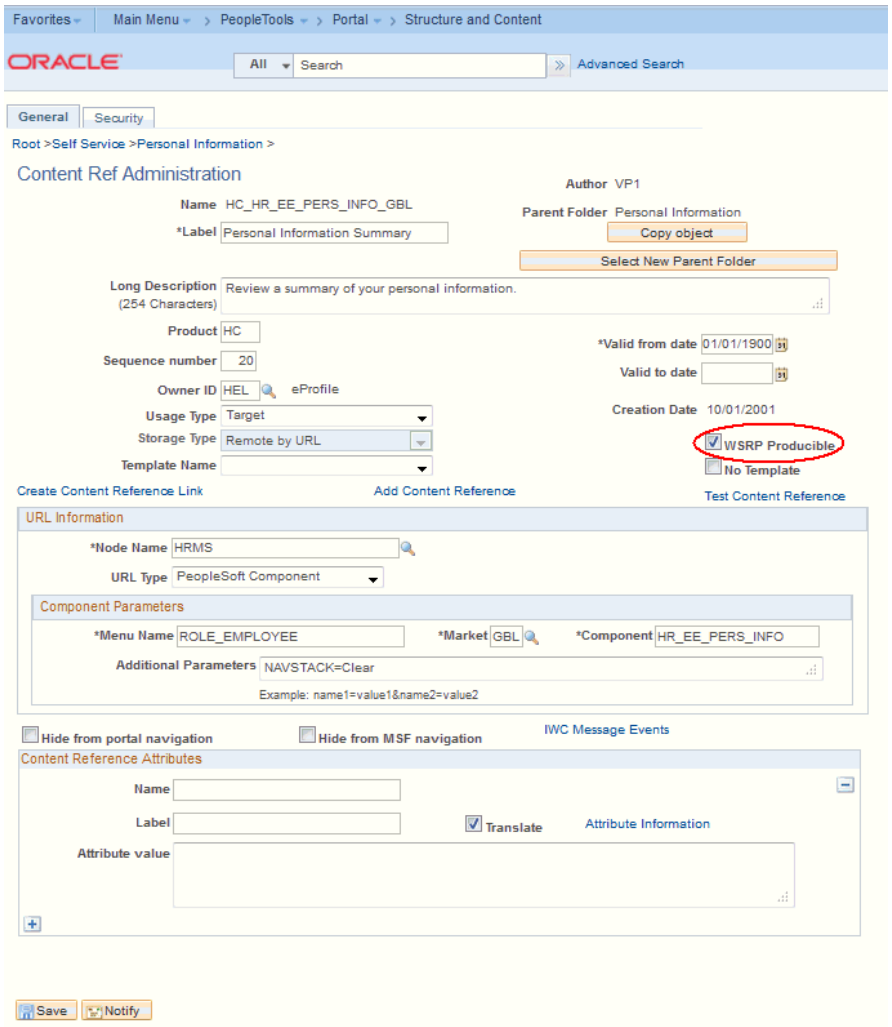
Figure 9-15 Structure and Content Page



5. Navigate to the folder and subfolder (if required) containing the service that you want to expose as portlet in WebCenter Portal and click **Edit** to open it. For example, you could select **Self Service**, **Personal Information**, and then **Personal Information Summary**.

   The Content Ref Administration page displays (see Figure 9-16).

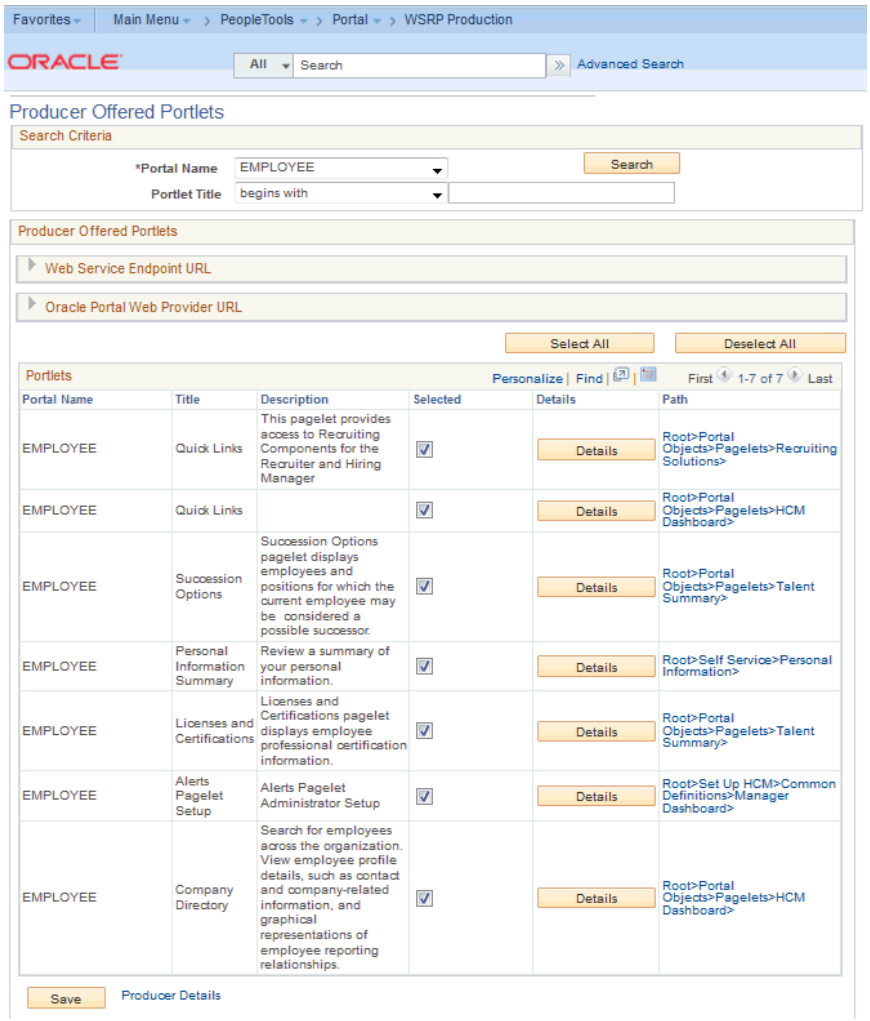
Figure 9-16 Content Ref Administration Page



6. On the General tab, select the **WSRP Producible** checkbox.
7. Save the page.
8. In the PeopleSoft Application Designer, open the component object to the Request Details page that gets displayed in PeopleSoft, and in the Component Properties section, check the **WSRP Compliant** check box.
9. From the main menu, expand PeopleTools and then Portal and select **WSRP Production**.

   The Producer Offered Portlets page displays (see Figure 9-17).

Figure 9-17 Producer Offered Portals Page



10. Verify that the service is exposed, and then expand Web Service Endpoint URL and copy the URL (the WSDL).
11. Open a new tab in your browser, and paste the copied URL into the Navigation Bar to access the WSDL page.
12. Copy the URL to the clipboard.
13. Continue by integrating the PeopleSoft WSRP producer in WebCenter Portal as described in How to Integrate PeopleSoft Applications in WebCenter Portal.

How to Configure WS-Security for PeopleTools 8.52 and Later

This section describes how to create a keystore for both WebCenter Portal and PeopleSoft, and exchange the private key between them. This step is required prior to adding WS-Security policies for WebCenter Portal.

1. First, we will create the WebCenter keystore as `webcenter.jks` with `orakey` as the private key, and PeopleSoft's public key `rootCA` and the certificate that PeopleSoft will use as the WS-Security recipient using the following `keytool` commands:

```
./keytool -genkeypair -keyalg RSA -dname "cn=orakey,dc=us,dc=oracle,dc=com" -alias
orakey -keypass password -keystore webcenter.jks -storepass password -validity 720
```

```
./keytool -exportcert -v -alias orakey -keystore webcenter.jks -storepass
password -rfc -file orakey.cer

./keytool -importcert -trustcacerts -alias orakey -file orakey.cer -keystore
peoplesoft.jks -storepass password
```

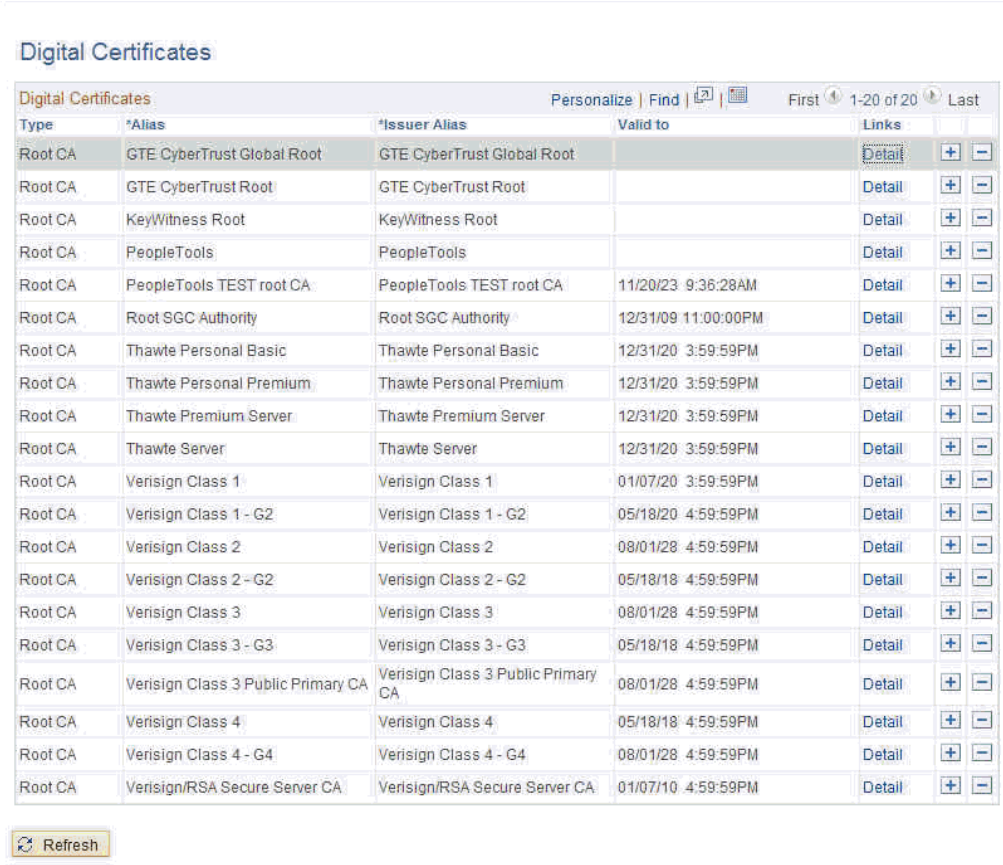
2. Next, we will create PeopleSoft keystore as `peoplesoft.jks` with `rootCA` as the private key and WebCenter's public key `orakey` and the certificate that WebCenter will use as the WS-Security recipient.

```
./keytool -genkeypair -keyalg RSA -dname "cn=rootCA,dc=us,dc=oracle,dc=com" -
alias rootCA -keypass password -keystore peoplesoft.jks -storepass password -
validity 720

./keytool -exportcert -v -alias rootCA -keystore peoplesoft.jks -storepass
password -rfc -file rootca.cer

./keytool -importcert -trustcacerts -alias rootCA -file rootca.cer -keystore
webcenter.jks -storepass password
```

3. After creating the key stores for WebCenter Portal and PeopleSoft, copy the `peoplesoft.jks` to the PeopleSoft host and `webcenter.jks` to the WebCenter host:
   - Copy `peoplesoft.jks` to `<Domain_Home>/config/fmwconfig/`
   - Copy `webcenter.jks` to `/home/psadm2/psft/pt/8.52/webserv/<Domain_Name>/applications/peoplesoft/pspc.war/WEB-INF/classes`
4. Install the certificate in PeopleSoft as shown below:
   a. Log into PeopleSoft as an administrator and navigate to **PeopleTools > Security > Security Objects > Digital Certificate**.

      The Digital Certificates page displays (see Figure 9-18).

Figure 9-18 Digital Certificates Page



Digital Certificates

Digital Certificates — Personalize | Find | First 1-20 of 20 Last

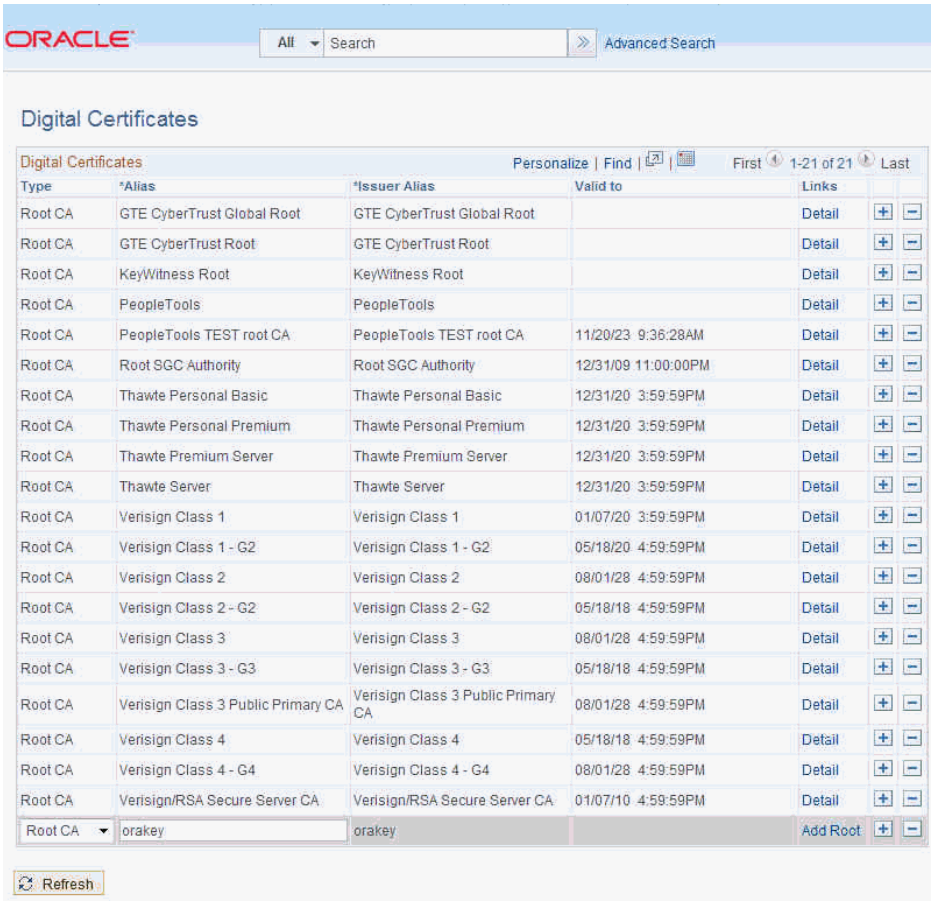
Type	*Alias	*Issuer Alias	Valid to	Links
Root CA	GTE CyberTrust Global Root	GTE CyberTrust Global Root		Detail
Root CA	GTE CyberTrust Root	GTE CyberTrust Root		Detail
Root CA	KeyWitness Root	KeyWitness Root		Detail
Root CA	PeopleTools	PeopleTools		Detail
Root CA	PeopleTools TEST root CA	PeopleTools TEST root CA	11/20/23 9:36:28AM	Detail
Root CA	Root SGC Authority	Root SGC Authority	12/31/09 11:00:00PM	Detail
Root CA	Thawte Personal Basic	Thawte Personal Basic	12/31/20 3:59:59PM	Detail
Root CA	Thawte Personal Premium	Thawte Personal Premium	12/31/20 3:59:59PM	Detail
Root CA	Thawte Premium Server	Thawte Premium Server	12/31/20 3:59:59PM	Detail
Root CA	Thawte Server	Thawte Server	12/31/20 3:59:59PM	Detail
Root CA	Verisign Class 1	Verisign Class 1	01/07/20 3:59:59PM	Detail
Root CA	Verisign Class 1 - G2	Verisign Class 1 - G2	05/18/20 4:59:59PM	Detail
Root CA	Verisign Class 2	Verisign Class 2	08/01/28 4:59:59PM	Detail
Root CA	Verisign Class 2 - G2	Verisign Class 2 - G2	05/18/18 4:59:59PM	Detail
Root CA	Verisign Class 3	Verisign Class 3	08/01/28 4:59:59PM	Detail
Root CA	Verisign Class 3 - G3	Verisign Class 3 - G3	05/18/18 4:59:59PM	Detail
Root CA	Verisign Class 3 Public Primary CA	Verisign Class 3 Public Primary CA	08/01/28 4:59:59PM	Detail
Root CA	Verisign Class 4	Verisign Class 4	05/18/18 4:59:59PM	Detail
Root CA	Verisign Class 4 - G4	Verisign Class 4 - G4	08/01/28 4:59:59PM	Detail
Root CA	Verisign/RSA Secure Server CA	Verisign/RSA Secure Server CA	01/07/10 4:59:59PM	Detail

Refresh

b. Click **+**to add a new entry.

We need to add digital certificates for `Remote` and `RootCA` as shown in Figure 9-19.

Figure 9-19 Digital Certificates Page



c. Enter the **Type** as `RootCA`, **Alias** as `orakey`, **Issuer Allias** as `orakey`, and then click the **Search** icon (magnifying glass).

d. Click **Import** and in the popup, enter the entire text of `orakey.cer` created earlier and click **OK**.

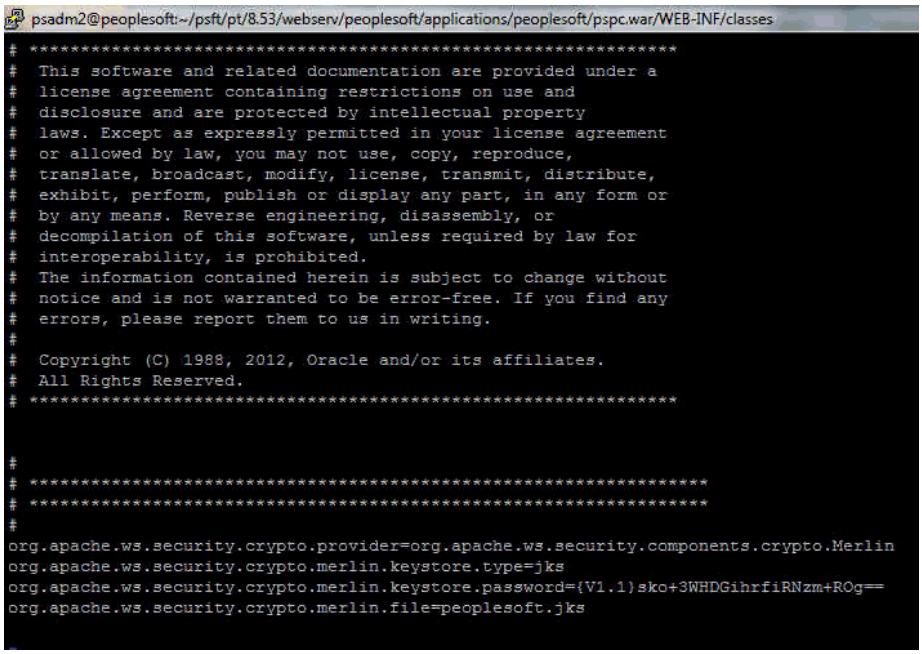
e. Click **+**to add another new entry, and enter the **Type** as `Remote`, **Alias** as `orakey`, **Issuer Allias** as `orakey` and then click the **Search** icon.

f. Click **Import** and in the popup, enter the entire text of `orakey.cer` created earlier and click **OK**.

5. Update the `WSS.properties` file under `/home/psadm2/psft/pt/8.52/webserv/<Domain_Name>/applications/peoplesoft/pspc.war/WEB-INF/classes` to reference the `peoplesoft.jks` file.

6. Use `PSCipher.sh` to create an Encrypted Password and update the KeyStore password as shown in Figure 9-20.

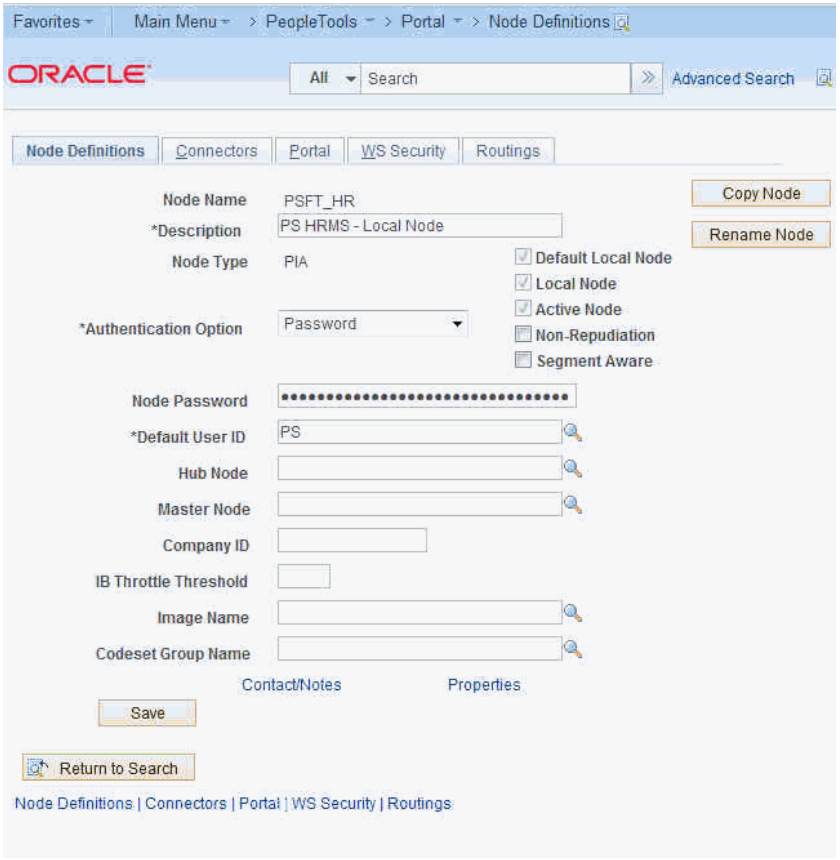
Figure 9-20 PSCipher.sh



7. Check the local node definition in PeopleSoft:
   a. Navigate to **Peopletools > Portals > Node Definitions**.

      The Nodes page displays.
   b. Click **Search** and click PSFT-HR.

Figure 9-21 Node Definitions Page



c. Select `Password` from the **Authentication Option** drop-down list and click **Save**.

8. Continue by adding a WS-Security policy to WebCenter Portal as described in How to Attach a WS-Security Policy to WebCenter Portal.

How to Attach a WS-Security Policy to WebCenter Portal

This section describes how to attach a WS-Security policy to WebCenter Portal.

> **Note:**
>
> Before continuing with the steps below you must have configured the WebCenter and PeopleSoft key stores as described in How to Configure WS-Security for PeopleTools 8.52 and Later.

- How to Configure WSS 1.0 SAML Token with Message Integrity
- How to Configure WSS 1.0 Username Token Without Password
- How to Configure WSS 1.0 SAML Token with Message Protection
- How to Configure WSS 1.0 Username Token with Password

How to Configure WSS 1.0 SAML Token with Message Integrity

Follow the steps below to configure the WSS1.0 SAML Token with Message Integrity policy for WebCenter Portal:

1. Navigate to the following directory on the PeopleSoft server:

   ```
   /home/psadm2/psft/pt/8.53/webserv/peoplesoft/piabin
   ```

   and run the following command:

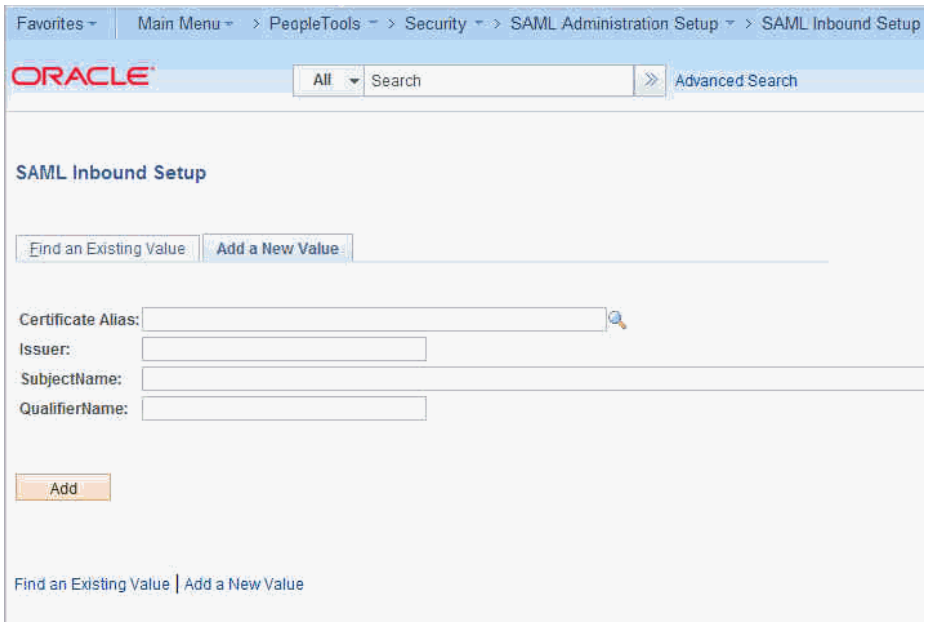
   ```
   ./redeployWSRP.sh 6
   ```

   This will update the PeopleSoft WSRP security options to use WSRPBaseService with SAMLToken Full Security.

2. In PeopleSoft, navigate to **PeopleTools > Security > SAML Administration Setup > SAML Inbound Setup**.

   The SAML Inbound Setup page displays (see Figure 9-22).

Figure 9-22 SAML Inbound Setup Page



3. Open the Add a New Value tab and map the WebCenter Portal user with the PeopleSoft user if they use a different OID (example settings are shown below), and then click **Save**.

   Example:

   - **Certificate Alias** - orakey
   - **Issuer** - WWW.ORACLE.COM
   - **SubjectName** - pat
   - **QualifierName** - WWW.ORACLE.COM
   - **Mapping PeopleSoft UserID** - PS

4. Continue by registering the WSRP producer and adding the portlet to a portal page as shown in How to Integrate PeopleSoft Applications in WebCenter Portal.

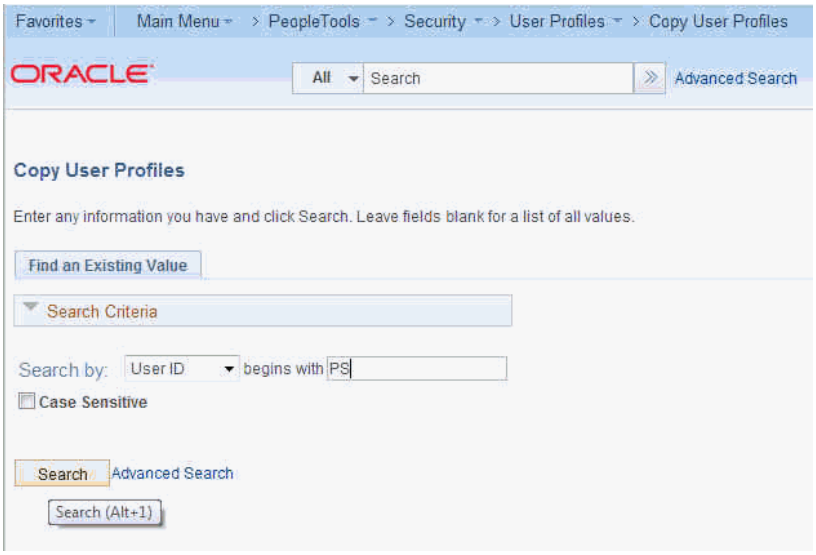
How to Configure WSS 1.0 Username Token Without Password

Follow the steps below to attach a WSS 1.0 Username Token without Password policy to WebCenter Portal.

1. Create a WebCenter user in PeopleSoft:
   a. In PeopleSoft, navigate to **PeopleTools > Security > User Profiles > Copy User Profiles**.

      The Copy User Profiles page displays (see Figure 9-23).

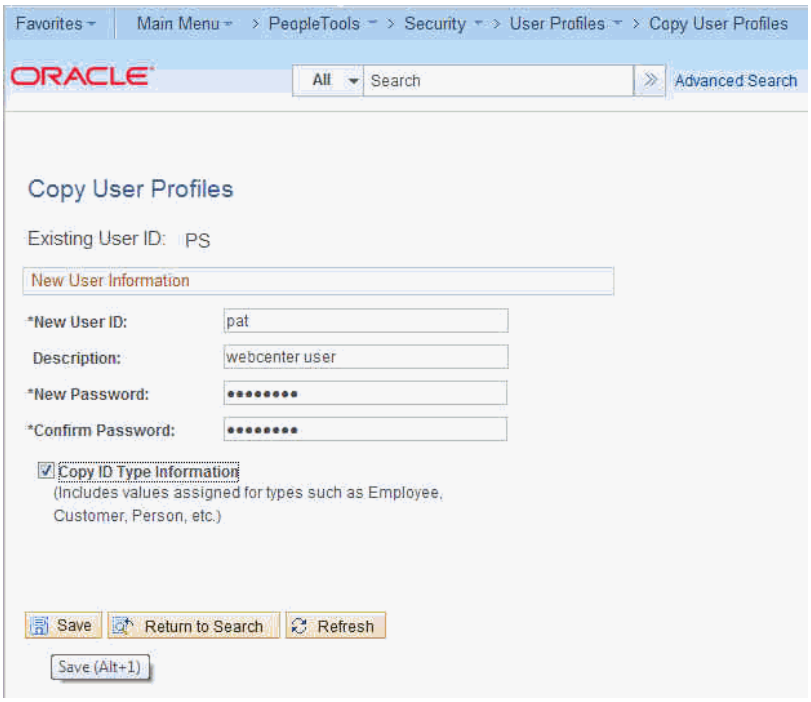
Figure 9-23 Copy User Profiles Page - Search Criteria



   b. Search for the user to add (`PS`, for example).

      The search results display (see Figure 9-24).

Figure 9-24 Copy User Profiles Page - Search Results



c. Enter the **New User ID** (for example, `Pat`), a **Description**, the **New Password**, check the **Copy ID Type Information** option and click **Save**.

2. Log into Fusion Middleware Control, select the domain and navigate to **Security > Security Provider Configuration**.

   The Security Provider Configuration page displays.

3. Open the Keystore section and click **Configure**.

   The Keystore Configuration page displays.

4. Enter `./webcenter.jks` for the **KeyStore Path**, `orakey` for the **Key Alias**, `orakey` for the **Crypt Alias**. Enter the associated passwords and click **OK**.

   Note that you must restart the entire domain for the configuration changes to take effect.

5. Navigate to `/home/psadm2/psft/pt/8.53/webserv/peoplesoft/piabin` and run the following command:

   ```
   ./redeployWSRP.sh 8
   ```

   This will update the PeopleSoft WSRP security options to use WSRPBaseService with UsernameToken, No Password Full Security Option With WSS Response.

6. Continue by registering the WSRP producer and adding the portlet to a WebCenter Portal or portal page as shown in How to Integrate PeopleSoft Applications in WebCenter Portal.

How to Configure WSS 1.0 SAML Token with Message Protection

Follow the steps below to attach the WSS1.0 SAML Token with Message Protection policy to WebCenter Portal.

1. Navigate to `/home/psadm2/psft/pt/8.53/webserv/peoplesoft/piabin` and run the following command:

```
./redeployWSRP.sh 10
```

This will update the PeopleSoft WSRP security options to use WSRPBaseService with SAMLToken Full Security Option With WSS Response.

2. Continue by registering the WSRP producer and adding the portlet to a portal page as shown in How to Integrate PeopleSoft Applications in WebCenter Portal.

How to Configure WSS 1.0 Username Token with Password

Follow the steps below to attach the WSS1.0 SAML Token with Message Protection policy to WebCenter Portal.

1. Navigate to `/home/psadm2/psft/pt/8.53/webserv/peoplesoft/piabin` and run the following command:

   ```
   ./redeployWSRP.sh 7
   ```

   This will update the PeopleSoft WSRP security options to use WSRPBaseService with UsernameToken Full Security Option With WSS Response.

2. Continue by registering the WSRP producer and adding the portlet to a WebCenter Portal or portal page as shown in How to Integrate PeopleSoft Applications in WebCenter Portal.

How to Integrate PeopleSoft Applications in WebCenter Portal

This section describes how to integrate a PeopleSoft application in WebCenter Portal.

This section contains the following subsections:

- How to Register the PeopleSoft WSRP Producer for WebCenter Portal
- How to Add the PeopleSoft Portlet to a WebCenter Portal Page
- How to Test the Portal Portlet Connection

How to Register the PeopleSoft WSRP Producer for WebCenter Portal

You can register the PeopleSoft WSRP producer directly in WebCenter Portal, as described in Registering Portlet Producers in *Administering Oracle WebCenter Portal*. You can also register the PeopleSoft WSRP producer using Fusion Middleware Control as described in the steps below.

To register the PeopleSoft WSRP producer using Fusion Middleware Control:

1. Prepare the PeopleSoft page that you want to consume in WebCenter Portal for remote access as described in How to Prepare the PeopleSoft Application for Remote Access.
2. Log into Fusion Middleware Control for the WebCenter Portal domain (`WC_Domain` by default).
3. Expand `WebCenter Portal` in the Navigation bar and from the WebCenter Portal menu, select **Register Producer**.

   The Add Portlet Producer page displays.
4. Set the **Producer Type** to `WSRP Producer`, enter a **Connection Name**, and paste the WSDL endpoint URL that you copied in step 1 into the **URL End Point** field.

5. If required, configure WS-Security in WebCenter Portal as described in How to Configure WS-Security for PeopleTools 8.51.
6. Click **OK** and verify that the producer connection was created successfully.
7. Continue by adding the portlet to a WebCenter Portal or portal page as described in How to Add the PeopleSoft Portlet to a WebCenter Portal Page.

How to Add the PeopleSoft Portlet to a WebCenter Portal Page

Follow the steps below to add the PeopleSoft portlet to a WebCenter Portal or portal page:

1. Log into WebCenter Portal.

   If you configured WS-Security, be sure to use the user account that was used in the SAML Inbound Setup page in PeopleSoft (see How to Attach a WS-Security Policy to WebCenter Portal).
2. Go to the page, or create a new page, where you want to add the PeopleSoft portal.
3. Click **Add Content** and in the resource catalog, select **UI Components** and then **Portlets**.

   Note that if you've created a custom catalog, **Portlets** may not appear. In this case, you will need to add it to the resource catalog. For information about managing resource catalogs, see Working with Resource Catalog Components on a Page in *Building Portals with Oracle WebCenter Portal*.
4. Click the portlet you added in Fusion Middleware Control.
5. Click **Add** for the PeopleSoft page you want to add to your portal page.
6. On the portal page, click the portlet's **View Actions** menu, and select **Display Options**.
7. In the Display Options dialog, set **Render Portlet in IFrame** to `True` and click **OK**.
8. Continue by checking the portlet connection as described in How to Test the Portal Portlet Connection.

How to Test the Portal Portlet Connection

Follow these steps to test the portlet connection by modifying content and checking that the modification shows up in the PeopleSoft application.

1. On the WebCenter Portal or portal page that you added the PeopleSoft portlet to, modify some information that you can verify the changes for in the PeopleSoft application.
2. Save your changes and confirm that the changes also appear in the PeopleSoft application.

How to Configure WS-Security for PeopleTools 8.51

This section describes the supported OWSM policies for PeopleTools 8.51. It is important to note that PeopleTools release 8.51 does not support outgoing WS-Security headers in its messages. However, some out-of-the-box Oracle WebCenter Portal/OWSM policies require that both outgoing and incoming messages be secured. To bridge this gap you may need to create custom OWSM policies. The different integration scenarios that would require you to create custom WS-Security policies, and the steps required on the WebCenter Portal side to configure them are also described in this section.

For integration scenarios with PeopleTools 8.51, you can use WSS10 SAML Token with Message Integrity, WSS10 SAML Token with Message Protection, or WSS10 Username Token with Password as the OWSM policy.

This section includes the following subsections:

- How to Configure WS-Security for WSS10 SAML Token with Message Integrity
- How to Configure WS-Security for WSS10 SAML Token with Message Protection
- How to Configure WS-Security for WSS10 Username Token with Password

How to Configure WS-Security for WSS10 SAML Token with Message Integrity

(PeopleSoft policy: `WSRPBaseService with SAMLToken Full Security Option (timestamp)`)

This section describes how to configure WS-Security for the WSS10 SAML Token with Message Integrity (`oracle/wss10_saml_token_with_message_integrity_client_policy`) policy.

To configure WS-Security:

1. Configure the Oracle WebCenter Portal/OWSM keystore as described in Configuring Web Services Security.
2. Generate a certificate containing the public key of the Oracle WebCenter Portal domain and send it to the PeopleTools administrator so it can be imported in the PeopleTools configuration.
3. When you register the producer, choose `wss10_saml_token_with_message_integrity_client_policy`.
4. Continue by adding the WSRP portlet to WebCenter Portal.

How to Configure WS-Security for WSS10 SAML Token with Message Protection

(PeopleSoft policy: `WSRPBaseService with SAMLToken Full Security Option (timestamp) With WSS Response`)

The default WSS10 SAML Token with Message Protection (`oracle/wss10_saml_token_with_message_protection_client_policy`) policy that ships with OWSM requires that response also be signed and encrypted. However, PeopleTools release 8.51 and earlier cannot send WS-Security headers in response (only the initial `cookie/get portlet handle` call contains security headers; subsequent calls do not) and we therefore need to create and attach a custom policy based on the `oracle/wss10_saml_token_with_message_protection_client_policy` policy.

To create a custom policy:

1. Log into Fusion Middleware Control and navigate to the Oracle WebCenter Portal domain (`WC_Domain` by default).
2. From the WebLogic Domain menu, select **Web Services > Policies**.
3. Select the `wss10_saml_token_with_message_protection_client_policy` and click **Create Like**.
4. Give the policy a new name (for example, `oracle/wss10_saml_token_with_message_protection_plaintext_response_client_policy`).

5. Open the Response tab, uncheck the **Include Entire Body** check boxes under Message Signing Setting and Message Encrypt Setting, and save the policy.
6. Check that the public certificate of the PeopleSoft keystore is imported into the keystore used in the WebCenter Portal domain.
7. Use WLST to register the producer using the newly created policy as shown in the following example:

```
registerWSRPProducer('webcenter', 'wc-pt851-saml_msg-protection', 'http://
xmlns.oracle.com/pspc/pswsdl/ps/EMPLOYEE', timeout=100, tokenType='oracle/
wss10_saml_token_with_message_protection_plaintext_response_client_policy',
enforcePolicyURI='false', issuer='www.oracle.com',
sigKeyAlias='webcenter',sigKeyPswd='welcome1', encKeyAlias='webcenter',
encKeyPswd='welcome1', recptAlias='peopleTools_public')
```

Use the alias for the imported `peoplesft` public key as the value for the `recptAlias` parameter.

> **Note:**
>
> You must use WLST to register the producer. Fusion Middleware Control can only accept fixed policy names and therefore you must register the producer with this policy using WLST by passing in `enforcePolicyURI='false'`.

How to Configure WS-Security for WSS10 Username Token with Password

(PeopleSoft policy: `WSRPBaseService with UsernameToken Full Security Option With WSS Response`)

The default WSS10 Username Token with Password `(oracle/wss10_username_token_with_message_protection_client_policy)` policy that ships with OWSM requires that response also be signed and encrypted. However, PeopleTools release 8.51 and earlier cannot send WS-Security headers in response (only the initial `cookie/get portlet handle` call contains security headers; subsequent calls do not) and we therefore need to create and attach a custom policy based on the `oracle/wss10_username_token_with_message_protection_client_policy` policy.

To create a custom policy:

1. Log into Fusion Middleware Control and navigate to the WebCenter Portal domain (`WC_Domain` by default).
2. From the WebLogic Domain menu, select **Web Services > Policies**.
3. Select the `wss10_username_token_with_message_protection_client_policy` and click **Create Like**.
4. Give the policy a new name (for example, `oracle/wss10_username_token_with_message_protection_plaintext_response_client_policy`).
5. Open the Response tab, uncheck the **Include Entire Body** check boxes under Message Signing Setting and Message Encrypt Setting, and save the policy.
6. Check that the public certificate of the PeopleSoft keystore is imported into the keystore used in the Oracle WebCenter Portal domain.

7. Use WLST to register the producer using the newly created policy as shown in the following example:

```
registerWSRPProducer('webcenter', '<Producer_Name>', '<URL>', timeout=100,
tokenType='oracle/
wss10_username_token_with_message_protection_plaintext_response_client_policy
', extApp='<Ext_App_Name>',
enforcePolicyURI='false', issuer='www.oracle.com',
sigKeyAlias='webcenter',sigKeyPswd='welcome1', encKeyAlias='webcenter',
encKeyPswd='welcome1', recptAlias='peopleTools_public')
```

Use the alias for the imported `peoplesft` public key as the value for the `recptAlias` parameter.

> **Note:**
>
> You must use WLST to register the producer. Fusion Middleware Control can only accept fixed policy names and therefore you must register the producer with this policy using WLST by passing in `enforcePolicyURI='false'`.

How to Integrate PeopleSoft Applications as Data Controls in WebCenter Portal

This section describes how to add PeopleSoft applications as Web service data controls in WebCenter Portal.

This section includes the following subsections:

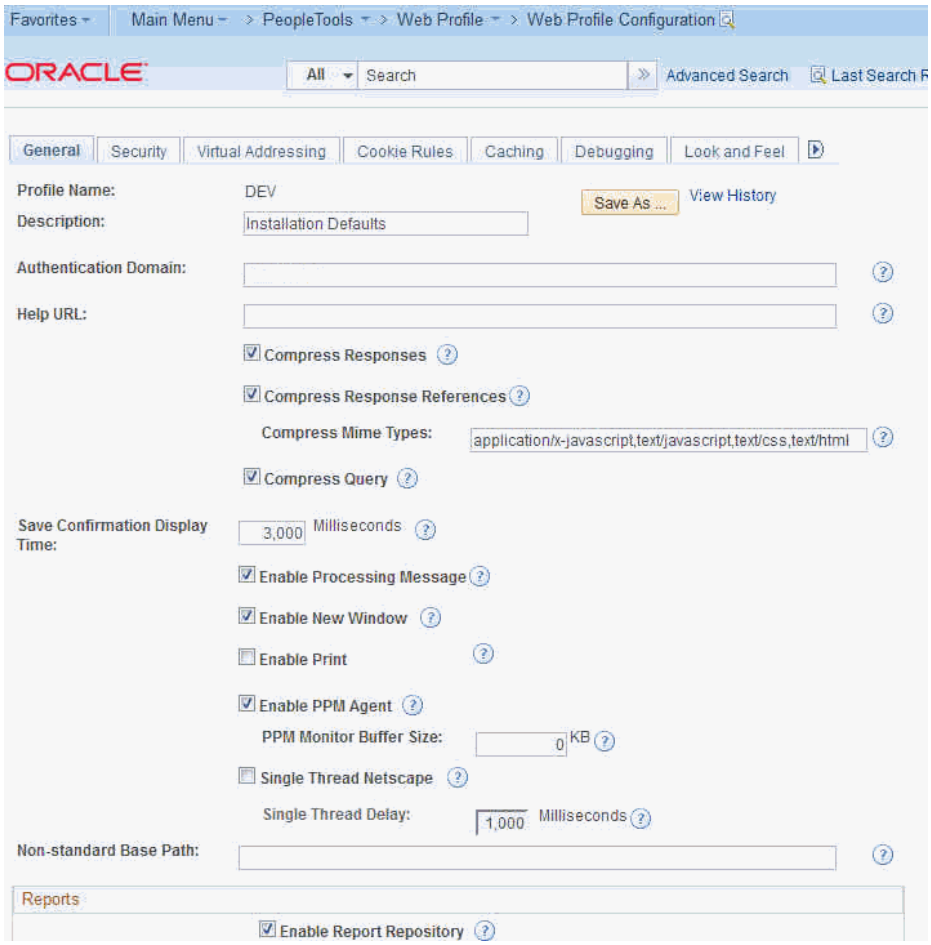
- How to Prepare the WSDL
- How to Create a Web Service Data Control

How to Prepare the WSDL

Follow the steps below to prepare a the WSDL.

1. Log into the PeopleSoft Console as an administrator.
2. Navigate to **PeopleTools > Web Profile > Web Profile Configuration**.
3. Click **Search** and select `DEV` from the results list.

Figure 9-25 WebProfile Configuration Page



4. Open the General tab and enter the **Authentication Domain** for your host.

   For example, if your host name is `ps.example.com`, enter `.example.com` in the **Authentication Domain** field.

5. Save your changes and close the application.
6. Open the `C:\Windows\System32\drivers\etc\hosts` file for editing.
7. On a new line enter the IP address and the full host name with the authentication domain.

   For example:

   ```
   193.128.1.113 ps.example.com
   ```

8. Save the file and reboot the server.
9. Log into the PeopleSoft application using the following URL:

   ```
   http://<host_name>:8000/ps/signon.html
   ```

   For example:

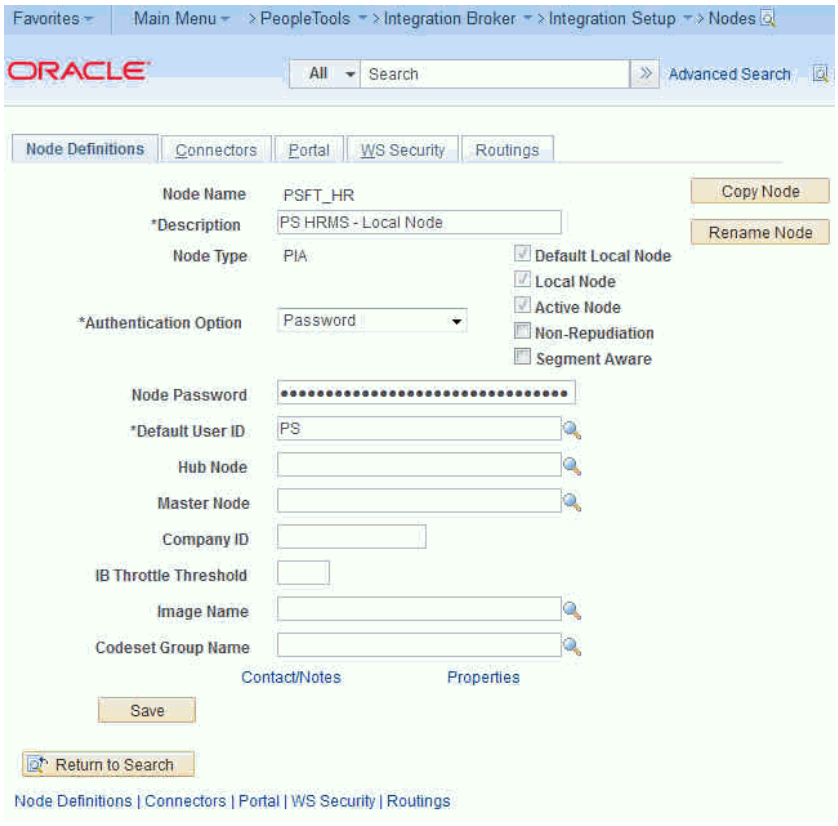
   ```
   http://ps.example.com:8000/ps/signon.html
   ```

10. From the Main Menu, navigate to **PeopleTools > Integration Broker > Configuration > Gateways**.
11. Search for the **GatewayID** `LOCAL`. The Local Gateway URL is set to

```
http://<host_name>:8000/PSIGW/PeopleSoftListeningConnector
```

12. Using the Local Gateway URL, ping the gateway to make sure it's active.
13. Open the Gateway Setup Properties and log in as an administrator.
14. On the PeopleSoft Node Configuration page, check that the node being used is `PSFT_HR`.
15. Ping the node
16. From the Main Menu, navigate to **PeopleTools > Integration Broker > Configuration > Service Configuration**.
17. Open **Setup Target Locations** and check that the **Target Location** is set to `<Local Gateway URL>/PSFT_HR`.
18. From the Main Menu, navigate to **PeopleTools > Integration Broker > Integration Setup > Nodes**.
19. Click **Search**.
20. Click the **Default Local Node** `PSFT_HR`.
21. On the Nodes tab, check that the **Default UserID** is set correctly as in the example in Figure 9-26.

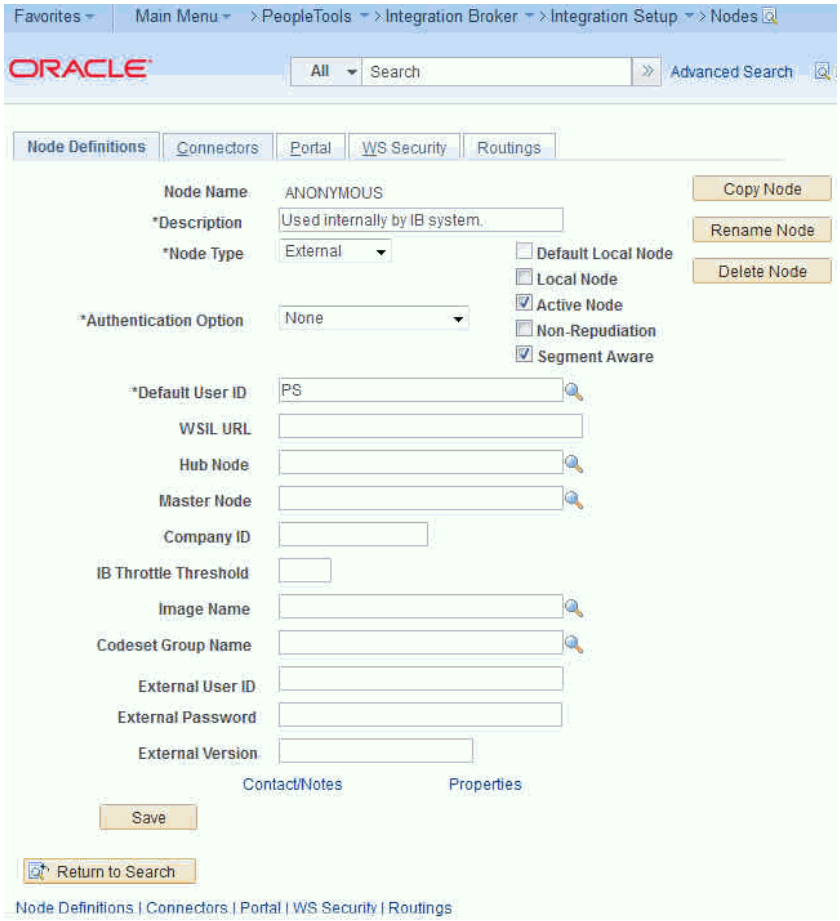
Figure 9-26 Nodes Page - Node Definitions



22. Click **Return to Search**.
23. Click the **ANONYMOUS** node.

24. Change the **Default UserID** to the PeopleSoft Login ID (for example, **PS**) as in the example in Figure 9-27.

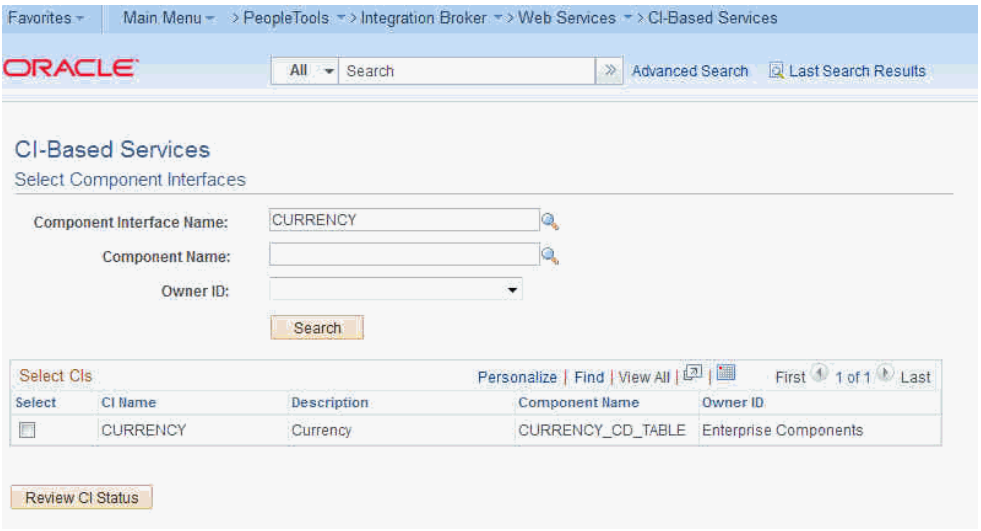
Figure 9-27 Nodes Page - Nodes Definitions



25. Save the changes and navigate to **Main Menu > PeopleTools > Integration Broker > Web Services > CI-Based Services**.

26. Search for and select the **Component Interface Name** (for example, `CURRENCY`) as in the example in Figure 9-28.

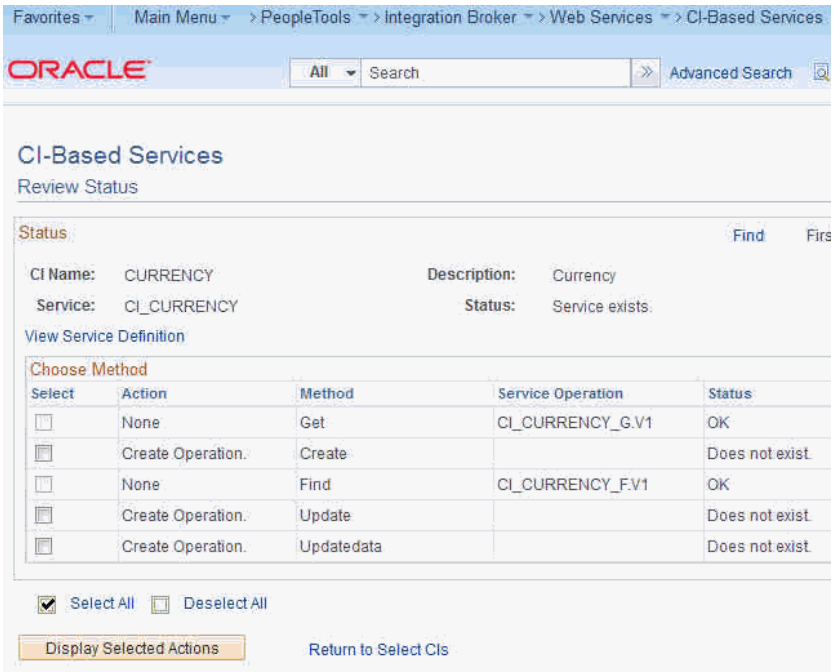
Figure 9-28 CI-Based Services Page - Select Component Interfaces



27. Click **Review CI Status**.

    The CI-Based Services - Review Status page displays (see Figure 9-29).

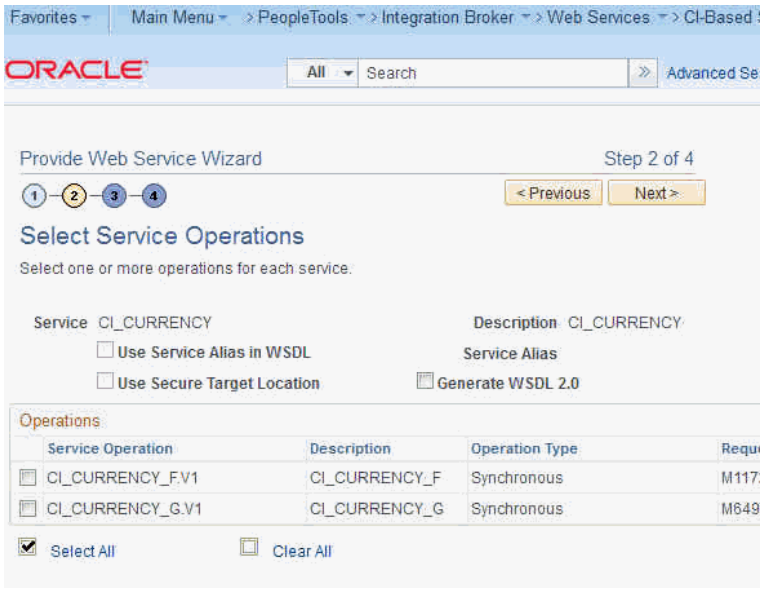
Figure 9-29 Review CI-Based Status - Review Status Page



28. Select the available methods (`Get` and `Find`, in this case) and click **Display Selected Actions**.
29. On the Confirm Actions dialog, click **Perform Selected** actions.
30. Click **View Service Definition**.
31. Click **Provide Web Service**.

    The Select Service Operations page displays (see Figure 9-30).

Figure 9-30 Select Service Operations Page



32. Select the **Select All** check box and click **Next** until you reach the last page.
33. Click **Finish** to generate the WSDL.

    You should now be able to access the WSDL URL. For this example, the URL would be:

    ```
    http://ps.example.com:8000/PSIGW/PeopleSoftServiceListeningConnector/
    PSFT_HR/CI_CURRENCY.1.wsdl\\\\
    ```

34. Continue by creating w Web service data control as shown in How to Create a Web Service Data Control.

How to Create a Web Service Data Control

Once you have the WSDL, you can continue by using it to create a Web service data control. In this section we'll continue with the example we started in How to Prepare the WSDL.

> **Note:**
>
> Before you can add a data control or task flow containing a data control to a portal page you must first have configured WS-Security for WebCenter Portal. For more information about configuring WS-Security, see Configuring Web Services Security in *Administering Oracle WebCenter Portal*.

For more information about creating a Web service data control, see *Creating a Web Service Data Control* in *Building Portals with Oracle WebCenter Portal*. For information about Web service data controls, see also Web Service Data Controls in *Building Portals with Oracle WebCenter Portal*.

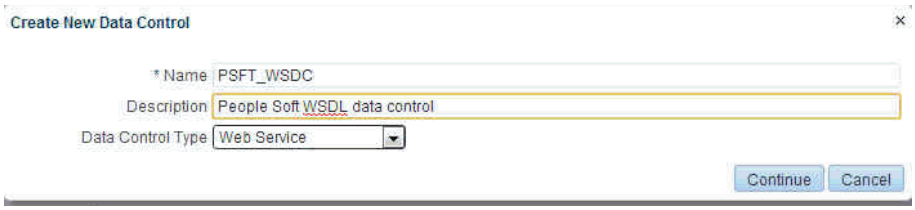
To create a Web service data control:

1. In WebCenter Portal or the portal in which you want to create the data control, go to either the **Shared Assets** or **Assets** page.

2. Select **Data Controls** and click **Create**.

   The Create New Data Control dialog displays (see Figure 9-31).

Figure 9-31 Create New Data Control Dialog

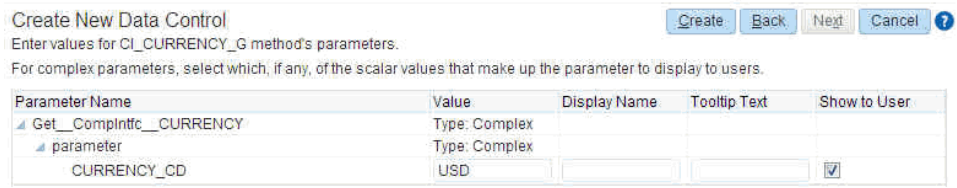


3. In the Create New Data Control dialog, enter a **Name** and **Description** for the data control, select `Web Service` as the **Data Control Type,** and then click **Continue**.
4. Enter the WSDL URL and other details for the data control and click **Continue**. For our example, the URL would be:

   ```
   http://ps.example.com:8000/PSIGW/PeopleSoftServiceListeningConnector/
   PSFT_HR/CI_CURRENCY.1.wsdl
   ```

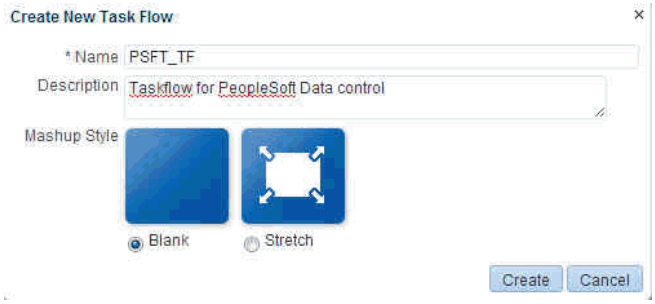
5. For our example, enter the Default Value for `CURRENCY_CD` as `USD` and click **Create** (see Figure 9-32).

Figure 9-32 Create New Data Control Dialog - CI_Currency_G Method Parameters



6. To make the data control available, from the **Shared Assets** or **Assets** page, select **Task Flows**. The Create New Task Flow dialog displays (see Figure 9-33).

Figure 9-33 Create New Task Flow Dialog



7. Click **Create** to create the task flow.

8. Select the task flow and click the **Edit** icon.
9. Add the data control (with parameter form) as a table onto the task flow and verify the data.
10. To make the task flow available, navigate to **Administration > Business Role Pages**.
11. Select **Business Role Page** and click the **Create** icon.
12. Edit the page. and save the changes.
13. Drop the task flow onto the page and verify the data.

Integrating Oracle Business Intelligence Presentation Services

This section explains how to configure WebCenter Portal to integrate with the Oracle Business Intelligence Presentation Services catalog. At runtime, users can add business intelligence objects to their WebCenter Portal pages.

This section includes the following subsections:

- About Integrating Oracle Business Intelligence Presentation Services
- How to Configure Credentials for Connecting to the Oracle BI Presentation Catalog
- How to Integrate Oracle Business Intelligence Objects in WebCenter Portal

About Integrating Oracle Business Intelligence Presentation Services

This section explains how to configure WebCenter Portal to integrate with the Oracle Business Intelligence Presentation Services catalog.

This section includes the following subsections:

- Understanding Oracle Business Intelligence Presentation Services Integration
- Requirements for Integrating Oracle Business Intelligence Presentation Services
- Advanced Integration Options

Understanding Oracle Business Intelligence Presentation Services Integration

Oracle WebCenter Portal users can expand and browse the Presentation Services catalog's folders to view an analysis' views. The following view types display in the Presentation Services catalog: table, pivot table, chart, funnel chart, gauge, narrative, ticker and title. The following view types do not display in the Presentation Services catalog: view selector, column selector, logical SQL, and no-results view.

Users can also browse the dashboard folder for the pages associated with the dashboard; however, users cannot browse within the dashboard pages to see their components (for example, any analyses embedded in the dashboard).

Requirements for Integrating Oracle Business Intelligence Presentation Services

You must also set up a connection to the BI application as well as configuring security as described in Creating an Oracle BI EE Presentation Services Connection in *Developer's Guide for Oracle Business Intelligence Enterprise Edition*. You will also need to specify the credentials for the connection, as described in How to Configure Credentials for Connecting to the Oracle BI Presentation Catalog.

The following prerequisites apply:

Oracle WebCenter Portal

- The `WC_Portal` server has been installed and configured, including the database connection, Content Server connection, and Fusion Middleware Control

OBIEE

- Oracle Business Intelligence Applications
- OBI Enterprise Edition version 12.2.1.1, 12.2.1.2, or 12.2.1.3

> **Note:**
>
> If WebCenter Portal is not fully integrated with OBIEE and only calls OBIEE reports as links, then any OBIEE 12.2.1.x version is supported.

- OBIEE is already installed, configured, and up and running (Database –OBI Enterprise Edition)
- OBI Applications is installed and set up and all content is available from the OBIEE environment (Optional)

Security

The OBIEE integration requires that the identity store user name population be the same across WebCenter and OBIEE. This can be done by either:

- Having WebCenter and OBIEE share the same identity store (recommended)
- Maintaining identical user names across separate WC and OBIEE identity stores

Advanced Integration Options

As well as the approaches to adding resources described in the subsections in Integrating Oracle Business Intelligence Presentation Services, such as adding business intelligence analyses, dashboards, and scorecard components that can be easily dropped onto a page, there are also options for using Web services and BI EE Logical SQL view objects to embed business intelligence data into an application. For more information about using Web services, see Introduction to Oracle Business Intelligence Web Services in *Integrator's Guide for Oracle Business Intelligence Enterprise Edition*. For more information about using the BI EE Logical SQL view object, see Using the Oracle BI EE Logical SQL View Object in *Developer's Guide for Oracle Business Intelligence Enterprise Edition*.

How to Configure Credentials for Connecting to the Oracle BI Presentation Catalog

At design time, you need to specify credentials to connect to the Oracle BI Presentation Catalog. These credentials are used to retrieve the list of business intelligence objects (for example, analyses, dashboards, and scorecard components) from the Oracle BI Presentation Catalog.

This process ensures that the login to the Presentation Server is the same as the current user of the application and any access checks are performed as the current user, and data is fetched as the current user. If the ADF page contains business

intelligence objects to which the user does not have access, the ADF page returns a message stating that the user does not have the proper permissions to access these objects.

Note that the **Perform impersonation** parameter should be set to `true` when security is enabled.

This section contains the following subsections:

- How to Check for the BIImpersonateUser
- How to Create the BIImpersonateUser
- How to Grant Permissions to BIImpersonateUser

How to Check for the BIImpersonateUser

Use the following steps to check if a BIImpersonateUser user already exists, and that the roles assigned to it are correct:

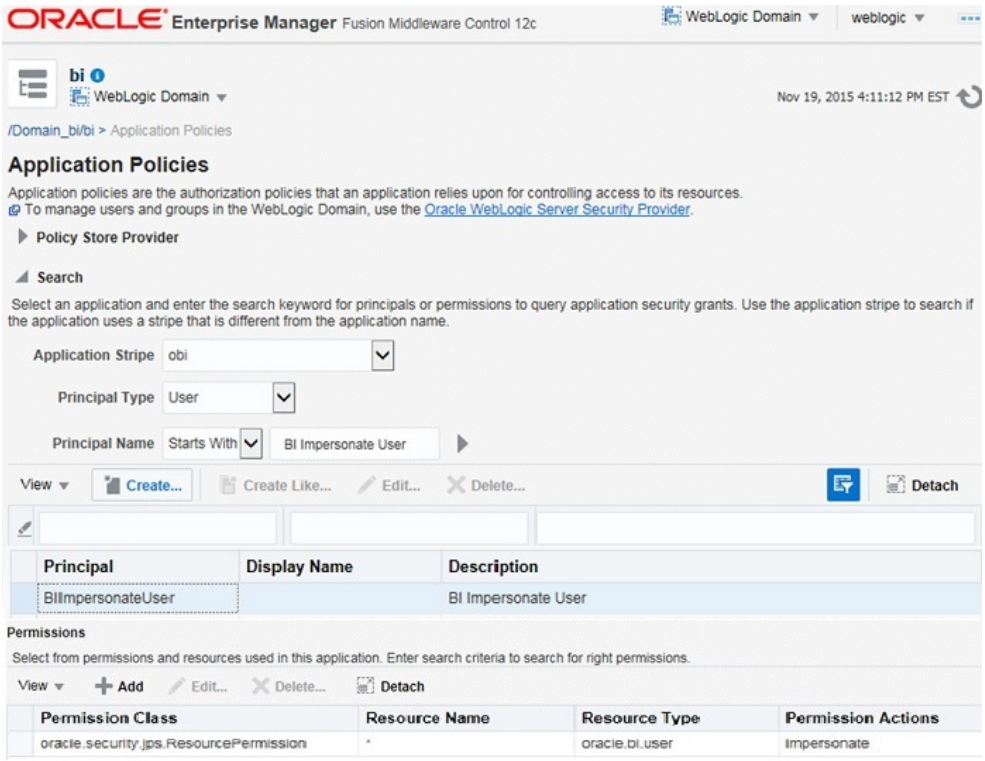
1. Open WLS Administration Console for your Oracle BI EE instance using an Administrator account.
2. Locate the Domain Structure pane and select Security Realm.

   The Realms pane displays.
3. In the Realms pane, select **<myrealm>**.

   The Settings dialog displays.
4. In the Settings dialog, open the Users and Groups tab.
5. Check that BIImpersonateUser appears in the list of users.

   If the BIImpersonateUser does not appear in the list, continue by creating the BIImpersonateUser as shown in How to Create the BIImpersonateUser.
6. Log into Fusion Middleware Control with an administrator account.
7. From the **Weblogic Domain** menu, select **Security > Application Policies**.
8. On the Application Policies page under Search, choose **obi** from the **Application Stripe** dropdown list.
9. From the **Principal Type** drop down list, select `User`.
10. In the **Name** field, enter `BIImpersonateUser` and start the search (Figure 9-34).

Figure 9-34 Application Policies Pane - bifoundation_domain



11. If found, check that:
    - **Resource Name** =`*`
    - **Resource Type** =`oracle.bi.user`
    - **Permission Actions = impersonate**
    - **Permission Class** =`oracle.security.jps.ResourcePermission`
12. If the BIImpersonateUser is not found, continue by adding permissions for the BIImpersonateUser as shown in How to Grant Permissions to BIImpersonateUser.

How to Create the BIImpersonateUser

Use the following procedures to create a BIImpersonateUser user to secure an application that uses an Oracle BI EE Presentation Services connection and includes Oracle BI EE objects. ADF security must be enabled for your application before you can apply the impersonator user credentials to the Oracle BI EE Presentation Services connection.

The Impersonate User feature secures applications that contain Oracle BI EE objects when Oracle BI EE and ADF are not sharing an Oracle Internet Directory (OID). Before you begin the process of creating and using Impersonate User, you must confirm that this capability is configured in your environment.

Before you perform this procedure, make sure that either you or the Administrator have created users in the WebLogic Server's Oracle BI EE realm and assigned the BIConsumer group to each user in this realm. For more information, see How to Create and Use Impersonate User in *Developer's Guide for Oracle Business Intelligence Enterprise Edition*.

Follow the steps below to create the BIImpersonateUser user:

1. Open WLS Administration Console for your Oracle BI EE instance using an Administrator account.
2. Locate the Domain Structure pane and select Security Realm.

   The Realms pane displays.
3. In the Realms pane, select **<myrealm>**.

   The Settings dialog displays.
4. In the Settings dialog, open the **Users and Groups** tab.
5. Confirm that the Users tab is displaying and click **New**.
6. Enter `BIImpersonateUser` for the user name and enter a password.
7. Click **OK**.

How to Grant Permissions to BIImpersonateUser

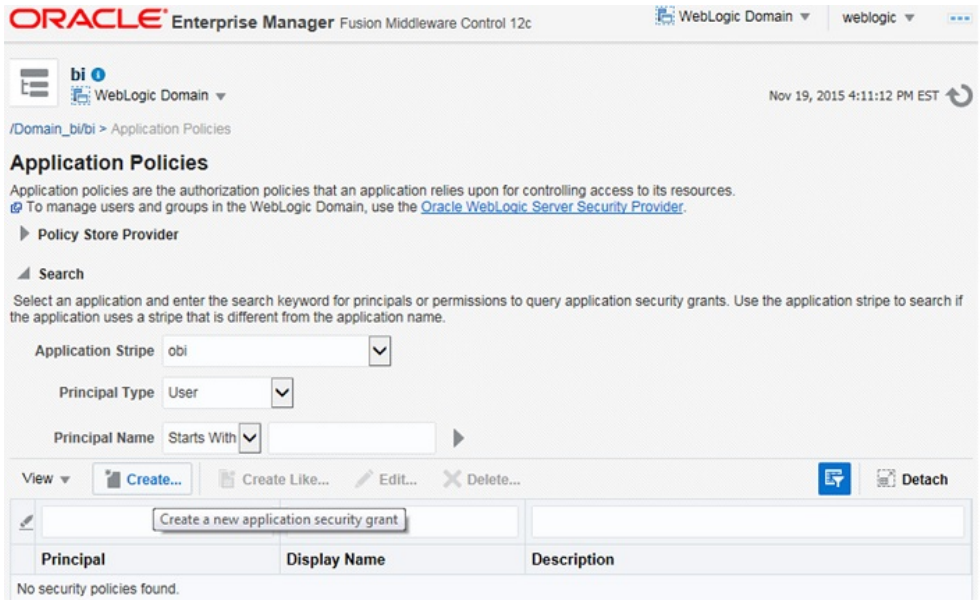
Follow the steps below to use Fusion Middleware Control to grant permissions to BIImpersonateUser:

1. From the **WebLogic Domain** drop down, select **Security > Application Policies**.

   The Search pane displays.
2. On the Application Policies page under Search, choose **obi** from the Application Stripe dropdown list. Set the **Principle Type** as **User**
3. Click **Create**.

   The Create Application Grant pane displays (Figure 9-35).

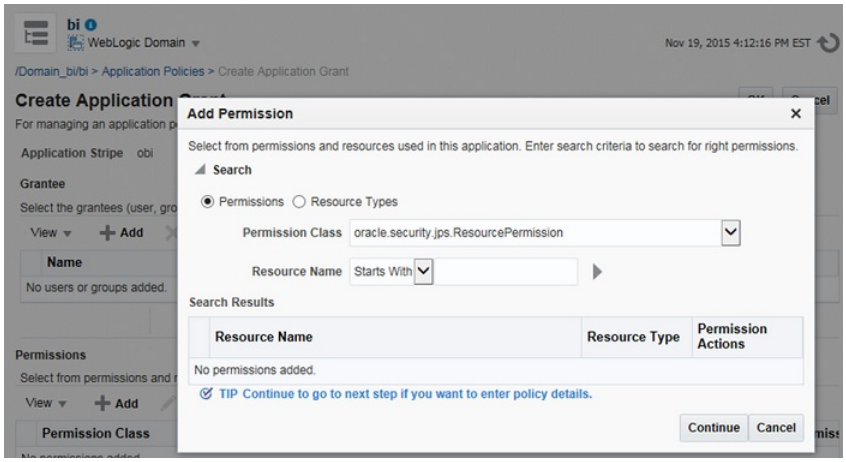
Figure 9-35 Create Application Grant Pane



4. Under **Permissions**, click **Add a new permission** and perform the following actions in the Add Permission pane.

The Add Permission dialog displays (Figure 9-36).

Figure 9-36 Add Permission Dialog



a. In the Permission Class list, select **oracle.security.jps.ResourcePermission**.

b. Select **Resource Types**.

c. In the Resource Type list, select **oracle.bi.user**, and click **Continue**.

d. In the Resource Name field, to impersonate all users, enter *.

e. For Permissions Actions, select **impersonate**, and then click **Select**.

Clicking **Select** will take you back to the Create Application Grant pane.

5. Under Grantee, click **Add users** and perform the following actions in The Add Principal pane:

   a. In the Type list, **User**.

   b. Click **Search Roles**.

   c. Select **BIImpersonateUser**, and click **OK** to return to the Create Application Grant pane.

6. In the Create Application Grant page, click **OK** to compete the creation of the security grant.

7. If the changes that you made do not display, stop and restart the following servers:

   - Oracle BI EE Server
   - Oracle BI EE Presentation Server
   - WebLogic Server

How to Integrate Oracle Business Intelligence Objects in WebCenter Portal

Use the following procedures to configure portal integration with the BI objects.

- How to Add or Modify a Presentation Services Connection After Deployment

- How to Add Oracle BI Objects to a WebCenter Portal Resource Catalog
- How to Add Oracle BI Content at Runtime
- How to Modify a Business Intelligence Object's Prompt Values
- How to Modify a Business Intelligence Task Flow's Initialization Parameters

How to Add or Modify a Presentation Services Connection After Deployment

Before you can begin integrating BI objects in WebCenter Portal, you must first configure a connection from WebCenter Portal to the BI server. Oracle BI EE provides an ADF MBean that lets you add a new connection to a deployed portal or BI ADF application. You can also modify a deployed application's existing connection. MBeans are deployed with the application and can be accessed post-deployment using Fusion Middleware Control.

Prior to following the steps below, you should already have followed the steps in How to Configure Credentials for Connecting to the Oracle BI Presentation Catalogto specify credentials to connect to the Oracle BI Presentation Catalog.

> **Note:**
>
> If the portal and the Oracle Business Intelligence application do not share the same identity store, you must create the relevant users in both systems.

Follow the steps below to configure the connection after the application was deployed.

1. Log into the **FMW Control Enterprise Manager** of the instance where **WebCenter Portal** is installed and click on the **WebLogic domain** drop down menu.
2. From the list, select **System MBean Browser**.

   The System MBean Browser pane displays.
3. In the System MBean Browser pane, navigate to the **ADF Connections** tree node by following the below path:
   a. Select the **Application Defined MBeans** tree node.
   b. Select the **oracle.adf.share.connections** tree node.
   c. Select the **Server: <my server name>** tree node.

      For example, `Server:DefaultServer` or `WC_Portal`.
   d. Select the **Application:<your application's name>** tree node.

      For example, `Application:Application2` or `webcenter`.
   e. Open the **ADF Connections** tree node.
   f. Open the child ADF Connections tree node.

      The corresponding MBean information displays in the Application Defined MBean pane.
4. In the Application Defined MBean pane, open the Operations tab and then click **createConnection** to create a Presentation Services connection.

   The Operation:createConnection dialog displays.
5. Specify the required values for the connection.

In the **Connection Type** value field, enter `BISoapConnection`, in the **Connection Name** value filed, enter for example, `biserver` and click **Invoke** to create the connection.

6. In the **System MBean Browser** pane, click **Refresh** to refresh the tree so that the new connection displays.
7. Continue to expand the tree **ADF Connections > BISoapConnection**. You should see the **biserver** connection that was created

   The connection's information displays in the Application Defined MBean pane.
8. Navigate to the **Attributes** tab.
9. Enter the `BISoap` connection information as shown below, and then click **Apply** to apply your changes.

```
context = analytics
host = The host name where the BI Server is running
IsStaticResourcesLocationAutomatic = true
Port = 9502(default Analytics port)
protocol = http
StaticResourcesLocation = http://machine.domain:port (This is the
default URL for the Analytics port.)
username =BIImpersonatorUser
password = BIImpersonatorUser user password
```

10. Keep the defaults for the rest of the fields and click **Apply** .
11. Click on the **ADFConnections** folder in the Navigation pane, open the **Operations** tab, and then click **Save** to save the connection.
12. When you click **Invoke**, you should get the following message:

    "`Confirmation Operation executed successfully.`"

How to Add Oracle BI Objects to a WebCenter Portal Resource Catalog

Before you can add Oracle BI content to a portal page, you must add objects stored in the Oracle BI Presentation Catalog to a WebCenter Portal resource catalog:

1. Log into WebCenter Portal as an administrator or application specialist.
2. In the portal browser, click the **Administration** tile.
3. Click **Shared Assets** , then **Resource Catalogs**.
4. Click **Create**.
5. In the **Name** field, enter the name of the resource catalog you are creating. Complete the other fields, as necessary.
6. Make the resource catalog available by selecting its **Available** check box.
7. Select the new resource catalog and click **Edit**.
8. From the **Add** menu, select **Add From Library**.
9. Double-click **Connections**.

   The BI Presentation Services folder displays.
10. Open this folder to display the Oracle BI objects and browse to and select the objects that you want to add.

11. Click **Add** to add the selected objects to the catalog.

> **Note:**
>
> If your portal does not render as expected in Chrome and Firefox, add the following parameter in the `instanceconfig.xml` file of the Oracle BI Presentation Server and restart the server:
> `<DisableContentSecurityPolicy>true</DisableContentSecurityPolicy>`
>
> If Oracle BI objects don't render as expected in Internet Explorer, update compatibility view settings to either remove the Oracle Business Intelligence web site from the list or disable the compatibility view mode.

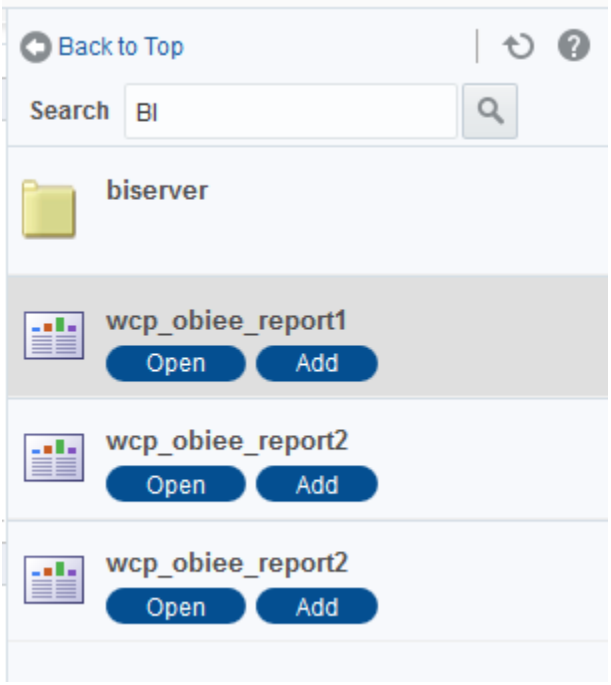
For more information about managing resource catalogs at runtime, see Working with Resource Catalogs in *Building Portals with Oracle WebCenter Portal*.

How to Add Oracle BI Content at Runtime

Use this procedure to create a portal page and add Oracle BI objects to it. Before you perform this procedure, you must have added Oracle BI objects to a WebCenter Portal resource catalog (see How to Add Oracle BI Objects to a WebCenter Portal Resource Catalog.

1. Log into WebCenter Portal and create a new portal or access an existing portal.
2. Make the resource catalog containing the Oracle BI objects the default resource catalog for portal pages. See Choosing Default Resource Catalogs
3. In the portal editor, create a new page.
4. In the page editor, click **Add Content** to open the resource catalog, and browse to the folder containing the Oracle BI objects.

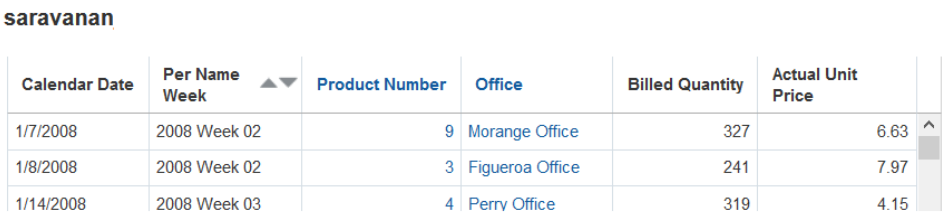
Figure 9-37 Resource Catalog with the Oracle BI Component



5. Select an analysis or dashboard and click **Add**.

   The object that you selected is added to the page.

Figure 9-38 Portal Page with Oracle BI Component



How to Modify a Business Intelligence Object's Prompt Values

Use this procedure to test the portal page by changing an analysis or dashboard's filter or prompt values.

1. Open the page that you created.
2. In the running page, click **Page Actions** and then click the **Edit** link to enter edit mode.
3. Add an analysis or dashboard that contains a filter or prompt. For more information about adding Oracle BI objects to the page, see How to Add Oracle BI Content at Runtime.
4. Without exiting the edit mode of the page, save the portal page.
5. In the portal page, modify the prompt values and click **OK**.

6. Exit edit mode, save the page and confirm that the application correctly applied the prompt values.

How to Modify a Business Intelligence Task Flow's Initialization Parameters

Use the following procedure to test the business intelligence task flow's initialization parameters.

1. Open the page that you created.
2. In the running page, click **Page Actions** and then click the **Edit** link to enter the edit mode.
3. Add an analysis or dashboard that is part of a task flow. For more information about adding business intelligence content to the `.jspx` page, see How to Add Oracle BI Content at Runtime.
4. Without exiting the edit mode for the page, save the portal page.
5. Locate the business intelligence object and click the **Edit** (wrench) icon.

   The Component Properties dialog displays.
6. On the portal page, open the **Parameters** tab and modify the object's parameters and click **OK**.
7. Open the Parameters tab, modify the object's parameters, and click **OK**.
8. Exit edit mode, save the page and confirm that the application correctly applied the modified parameter values.

Integrating with Oracle Content and Experience Cloud

This section describes how to integrate Oracle Content and Experience Cloud with WebCenter Portal.

It contains the following topics:

- About Oracle Content and Experience Cloud Integration
- Integrating Oracle Content and Experience Cloud with WebCenter Portal
- Creating a Default Oracle Content and Experience Cloud Connection Using WLST

About Oracle Content and Experience Cloud Integration

The Oracle DOCS Content Manager task flow enables users to integrate Oracle WebCenter Portal with Oracle Content and Experience Cloud and quickly access documents in Oracle Content and Experience Cloud. You can add, view, manage, and share documents and collaborate in cloud server with robust security from WebCenter Portal.

The Oracle DOCS Content Manager task flow:

- Provides easy and secure access to Oracle Content and Experience Cloud from WebCenter Portal
- Allows easy collaboration with other people

Integrating Oracle Content and Experience Cloud with WebCenter Portal

To integrate Oracle Content and Experience Cloud with WebCenter Portal:

1. Add the WebCenter domain in Oracle Content and Experience Cloud to allow the display of embedded content from Oracle Content and Experience Cloud within WebCenter Portal.

   > **Note:**
   >
   > Log in to Oracle Content and Experience Cloud as an administrator. From the user menu, select **Administration**, then **Documents**. On the Documents page, enable the **Embedded Content** option. In the **Allowed Domains** field, provide portal host name and port number.
   >
   > For example: *`hostname:port`*
   >
   > For more information, see Embedding Content in Other Domains in *Administering Oracle Content and Experience Cloud*.

2. Register Oracle Content and Experience Cloud with WebCenter Portal. For more information, see Creating a Default Oracle Content and Experience Cloud Connection Using WLST.
3. Add the Oracle DOCS Content Manager task flow to a portal page. For more information, see Adding the Oracle DOCS Content Manager Task Flow to a Page in *Building Portals with Oracle WebCenter Portal*.
4. (Optional) Customize the Oracle Content and Experience Cloud connection. For more information, see Customizing the Oracle Content and Experience Cloud URL Connection in *Building Portals with Oracle WebCenter Portal*.

   > **Note:**
   >
   > You can customize the Oracle Content and Experience Cloud connection to change the default URL connection to a different connection.

Creating a Default Oracle Content and Experience Cloud Connection Using WLST

Create a default Oracle Content and Experience Cloud URL connection in Oracle WebCenter Portal using the following WLST command:

```
adf_createHttpURLConnection(appName='application name',
name='connection name', url='http://host:port/documents')
```

where:

- `appName` is the application name. For example: `webcenter`.

- `name` is the default URL connection. Set this value to `WCP-DCS`.
- *`host`*`:`*`port`* is the host and the port of your Oracle Content and Experience Cloud.

Example:

```
 adf_createHttpURLConnection(appName='webcenter', name='WCP-DCS',
url='http://myhost:1080/documents')
```

For more information on WLST, see Oracle WebLogic Scripting Tool (WLST).

You can list the created connection and also delete the connection using the following WLST commands:

- To list the connections created, use:

  ```
  adf_listURLConnection(appName='application name')
  ```

  For example:

  ```
  adf_listURLConnection(appName='webcenter')
  ```

- To delete the connection, use:

  ```
  deleteConnection(appName='application name', name='connection name')
  ```

  For example:

  ```
  deleteConnection(appName='webcenter', name='WCP-DCS')
  ```

10

Configuring Search in WebCenter Portal

Configure search to index and search objects in WebCenter Portal.

> **Note:**
>
> Beginning with Release 12c (12.2.1.4.0), Oracle WebCenter Portal has deprecated the support for Oracle SES. If you have upgraded from a prior release, your upgraded instance might be configured to use Oracle SES. In this case, you must configure WebCenter Portal to use Elasticsearch or OCI Search Service with OpenSearch to index and search objects.

> **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 granted through WebCenter Portal Administration.
>
> For more information about roles and permissions, see Understanding Administrative Operations, Roles, and Tools.

Topics:

- Integrating with Elasticsearch
- Integrating with OCI Search Service with OpenSearch
- Creating a Crawl Admin User in WebCenter Portal
- Configuring WebCenter Content for Search
- Configuring WebCenter Portal for Search
- Synchronizing Users in WebCenter Portal
- Configuring Search Crawlers
- Configuring Search Custom Attributes
- Creating Custom Facets
- Customizing Search Query Pattern
- Scheduling a Crawl
- Customizing Search Settings in WebCenter Portal Administration
- Troubleshooting Issues with Elasticsearch
- Modifying the Default Connection Settings for Document Content Crawl Plugin in Elasticsearch Server
- Modifying Search Global Attributes

Integrating with Elasticsearch

Configure Elasticsearch to index and search objects in WebCenter Portal.

Topics:

- Understanding Search with Elasticsearch
- Prerequisites for Configuring Elasticsearch
- Configuration Roadmap for Elasticsearch in WebCenter Portal
- Installing Elasticsearch and Plugins

Understanding Search with Elasticsearch

Elasticsearch is a highly scalable search engine. It allows you to store, search, and analyze big volumes of data quickly and provides a distributed, multitenant-capable full-text search engine with an HTTP web interface and schema-free JSON documents.

Advantages of Elasticsearch

- Elasticsearch provides full-text search capabilities as it is built on Lucene.
- Elasticsearch is document-oriented. It stores data as structured JSON documents and indexes all fields by default, with a higher performance result.
- Elasticsearch is API driven; actions can be performed using a simple Restful API.
- Elasticsearch retrieves search results fast because it searches an index instead of searching the text directly.

You can configure Elasticsearch to search the following resources in WebCenter Portal:

- Documents, including wikis and blogs
- Portals, page metadata, page content (contents of HTML, Text, and Styled Text components), lists, and people resources
- Announcements and Discussions (available only for portals upgraded from prior releases)

Prerequisites for Configuring Elasticsearch

Ensure the following requirements:

- Oracle WebCenter Portal is installed.
- Optional. If you choose to use WebCenter Content for search, ensure that WebCenter Content is configured and all required components are enabled. See Managing Connections to Oracle WebCenter Content Server.

Configuration Roadmap for Elasticsearch in WebCenter Portal

Table 10-1 Roadmap - Setting Up Elasticsearch in WebCenter Portal

Actor	Task
Administrator	Creating a Crawl Admin User in WebCenter Portal
Administrator	Installing Elasticsearch and Plugins
Administrator	Configuring WebCenter Content for Search
Administrator	Configuring WebCenter Portal for Search
Administrator	Configuring Search Crawlers
Administrator	(Optional) Customizing Search Settings in WebCenter Portal Administration
Administrator	(Optional) Modifying Search Global Attributes
Administrator	(Optional) Configuring Search Custom Attributes
Administrator	(Optional) Creating Custom Facets

Installing Elasticsearch and Plugins

Elasticsearch can be installed as either single server set up or a cluster set up of minimum three servers.
The following topics describe how to set up a single server, a cluster, and use Oracle HTTP Server to do the load balancing:

- Setting Up Single Server
- Setting Up Cluster and Load Balancing Using OHS
- Securing Elasticsearch Using HTTPS

Setting Up Single Server

To install a single server setup, first you need to download the Elasticsearch 7.17.x binaries from Elasticsearch Releases. Then, follow the steps given below:

1. Set the following environment variables:
   - `JAVA_HOME` - Set the value to the Java home directory.
   - `PATH` - Update path to `$JAVA_HOME/bin`.
2. Edit the *`ORACLE_HOME`*`/wcportal/es/installES.properties` file and specify the values for the properties.

```
# A valid ORACLE_HOME. This must contain elasticsearch binaries at $ORACLE_HOME/
wcportal/es.
ORACLE_HOME=
#
# A valid Admin Server host which contains WebCenter Portal.
ADMIN_SERVER_HOST_NAME=
#
# Admin server port.
ADMIN_SERVER_PORT=
#
# A valid Weblogic administrator user name.
WLS_ADMIN_USER=
```

```
#
# A valid user name configured with crawler privileges.
SEARCH_APP_USER=
#
# Fusion Middleware configuration directory. This will typically
be $WEBCENTER_PORTAL_DOMAIN_HOME/config/fmwconfig.
# If $WEBCENTER_PORTAL_DOMAIN_HOME is not accessible from this host, copy
the directory $WEBCENTER_PORTAL_DOMAIN_HOME/config/fmwconfig
# from WebCenter Portal domain host to this host and provide that location.
WCP_FMW_CONFIG_LOCATION=
#
# Name of the Elasticsearch cluster. Default value is CURRENT_HOST_NAME and
it will be replaced with current hostname. This default is appropriate for
single-host environment only.
# If this value is changed, make sure that you don't reuse the same cluster
names in different environments. Otherwise, you might end up with nodes
joining the wrong cluster.
# Cluster name should be environment-specific and unique across your
enterprise.
ELASTIC_SEARCH_CLUSTER_NAME=CURRENT_HOST_NAME
#
# Name of the Elasticsearch node. Default value is CURRENT_HOST_NAME and it
will be replaced with current hostname. This default is appropriate for
single-host environment only.
# If this value is changed, make sure that you don't reuse the same node
name within the Elasticsearch cluster.
ELASTIC_SEARCH_NODE_NAME=CURRENT_HOST_NAME
#
# HTTP port of Elasticsearch server. Default value is 9200.
ELASTIC_SEARCH_PORT=9200
#
# Elasticsearch network host. Default value is 0.0.0.0 and it means
Elasticsearch server will listen on all network interfaces.
# If you have a requirement to limit traffic on multi-homed hosts with
multiple network interfaces, set to a specific hostname, IP address, or
network interface name.
# If network interface name is to be specified, it should be specified as:
_[networkInterface]_, for example _eth0_.
ELASTIC_SEARCH_NETWORK_HOST=0.0.0.0
#
# Elasticsearch home folder's parent directory. Configure the folder path
where the esHome folder should be created.
# Default value is DEFAULT and it means esHome will be created
in $ORACLE_HOME folder's parent directory.
# For example, if $ORACLE_HOME is /u01/oracle/fmw, then esHome will be /u01/
oracle/esHome.
ELASTIC_SEARCH_HOME_DIR_PARENT=DEFAULT
#
# Transport port of Elasticsearch server. Default value is 9300.
# This configuration is used for internal communication between nodes within
the cluster.
ELASTIC_SEARCH_TRANSPORT_PORT=9300
#
# An absolute path of Elasticsearch installer file. The installer can be
downloaded from- https://www.elastic.co/downloads/past-releases#elasticsearch
# The installer configured should be as per the OS where Elasticsearch
server is being installed (i.e. .tar.gz for Linux and .zip for Windows).
# For eg. for Elasticsearch 7.17.0 version this would be
elasticsearch-7.17.0-linux-x86_64.tar.gz on Linux and elasticsearch-7.17.0-
windows-x86_64.zip on Windows.
ELASTIC_SEARCH_INSTALLER_LOCATION=
```

```
#
# Elasticsearch version. The value should be corresponding to the installer file
which was downloaded.
ELASTIC_SEARCH_VERSION=
#
################################################################################
################################################################################
####
#
# NOTE: The following properties are required/applicable only if Elasticsearch is
being installed in windows.
#
################################################################################
################################################################################
####
#
# An absolute path of the directory where curl for Windows has been installed. The
curl for Windows can be downloaded from- https://curl.se/download.html
WINDOWS_CURL_HOME=
#
################################################################################
################################################################################
####
#
# NOTE: The following properties are required/applicable only if setting up a
cluster.
#
################################################################################
################################################################################
####
#
# Elasticsearch cluster discovery host list. All the hosts that will be part of
the cluster should be configured in this list.
# Each host should be declared by the appropriate fully-qualified hostname.
# Each host should be formatted in double-quotes separated with a comma and space.
The list should be wrapped in square-brackets. For example, ["host1.example.com",
"host2.example.com"].
ELASTIC_SEARCH_CLUSTER_HOST_LIST=
#
# Elasticsearch cluster discovery node list. All the nodes that will be part of
the cluster should be configured in this list.
# Each node should be declared by the corresponding value of
ELASTIC_SEARCH_NODE_NAME specified during the installation.
# Each node should be formatted in double-quotes separated with a comma and space.
The list should be wrapped in square-brackets. For example, ["node-1", "node-2"].
ELASTIC_SEARCH_CLUSTER_NODE_LIST=
#
# The name of the certificate which will be generated and used for the secure
communication between the nodes of the cluster.
# The certificate generated is of PKCS#12 format. Hence the certificate extension
should be .p12 only.
ELASTIC_SEARCH_CERTIFICATE_NAME=
#
# The primary host of Elasticsearch cluster. This should be one of the hosts in
ELASTIC_SEARCH_CLUSTER_HOST_LIST.
# Only during installation on this host, the certificate specified in
ELASTIC_SEARCH_CERTIFICATE_NAME and user specified in SEARCH_APP_USER are created.
# The configured value should be fully-qualified hostname.
ELASTIC_SEARCH_PRIMARY_HOST=
```

3. Enter the following command to install Elasticsearch and plugins:

```
ORACLE_HOME/oracle_common/common/bin/wlst.sh
ORACLE_HOME/wcportal/es/installES.py
ORACLE_HOME/wcportal/es/installES.properties
```

The following is the sample output for the successful installation on Linux:

```
...

-> Installing file:////tmp/webcenter-search-es-content-crawl-plugin.zip
-> Downloading file:////tmp/webcenter-search-es-content-crawl-plugin.zip
[==================================================] 100%
-> Installed orcl-wcp-es-content-crawl-plugin
-> Please restart Elasticsearch to activate any plugins installed
Creating the user crawladmin in Elasticsearch server...

...

The user crawladmin is created successfully.
Executing /home/Oracle/product/esHome/startElasticsearch.sh
Elasticsearch server installation is complete. Elasticsearch server home is
at: /home/Oracle/product/esHome
```

4. If Elasticsearch is being installed on Windows, execute the following command to secure the Elasticsearch server:

```
ORACLE_HOME/wcportal/es/secureESNode.cmd
ORACLE_HOME/wcportal/es/installES.properties
```

The following is the sample output for the successful installation:

```
...

"**************** Creating the user crawladmin in Elasticsearch server...
****************"
Enter password for user 'crawladmin': ********
"Please input the admin user password when prompted for host password below."
Enter host password for user 'elastic':
"Create user crawladmin is successful."
"**************** Stopping Elasticsearch server ****************"
"**************** Starting Elasticsearch server, please wait...
****************"
"************ Elasticsearch server setup completed successfully.
Elasticsearch server home is at: D:\Oracle\Middleware\esHome ************"
```

5. After the successful installation, verify that the Elasticsearch is configured properly.

   a. Access the URL `http://host:9200` using the Mozilla Firefox browser.

      A basic authentication dialog is displayed to enter the user name and password.

   b. Enter your crawl user name and password that you created in WebCenter Portal. See Creating a Crawl Admin User in WebCenter Portal.

      Successful login confirms that Elasticsearch Server is configured properly.

> **Note:**
>
> If you are unable to login or if Elasticsearch installation fails, check the logs in the following location:
>
> `/home/Oracle/product/esHome/esNode/logs`

Setting Up Cluster and Load Balancing Using OHS

In order to use the Elasticsearch cluster with WebCenter Portal, first you need to setup up the Elasticsearch cluster (there should be a minimum of three Elasticsearch servers) and then you can use Oracle HTTP Server (OHS) to do the load balancing in the Elasticsearch cluster. In the case of a cluster setup, some system settings of the host should be modified before installing the Elasticsearch server.

Modify System Settings

By default, the Elasticsearch server binds to loopback addresses for HTTP and transport (internal) communication. To join a cluster, an Elasticsearch server must be reachable via transport communication and bind transport to a non-loopback address. When bound to a non-loopback address, the Elasticsearch server enforces the bootstrap checks. Hence to succeed, the following system settings must be modified as a `root` user:

1. The maximum number of open file handles for the `<install-user>` should be at least 65,536. The current value for the logged-in user is obtained using the following command: `ulimit -n`. To increase it to the desired value, add the following line to the `/etc/security/limits.conf` file:

   ```
   <install-user>  -  nofile  65536
   ```

   Close and save the file. For the updated settings to take effect, the `<install-user>` should log out and log in again.

2. The maximum number of threads for the `<install-user>` should be at least 4096. The current value for the logged-in user can be obtained using the command: `ulimit -u`. To increase it to the desired value, add the following line to the `/etc/security/limits.conf` file:

   ```
   <install-user>  -  nproc  4096
   ```

   Close and save the file. For the updated settings to take effect, the `<install-user>` should log out and log in again.

3. The maximum virtual memory areas for the `<install-user>` should be at least 262144. The current value can be obtained using the command: `/sbin/sysctl vm.max_map_count`.
   To increase it to the desired value, add (or update) the following in the `/etc/sysctl.conf` file:

   ```
   vm.max_map_count=262144
   ```

Close and save the file. For the updated settings to take effect, run the following command: `/sbin/sysctl -p` as a `root` user.

Install Elasticsearch on Primary Host

Once the system settings are modified, Elasticsearch server should be installed with cluster specific configurations. Before proceeding with the installation, you need to identify the primary host of the cluster. The primary host is nothing but one of the hosts of the cluster where Elasticsearch server should be installed first. The procedure to install Elasticsearch server on the primary host is as follows:

Once the system settings have been modified, the Elasticsearch server should be installed with some specific configurations as mentioned below:

1. Edit the `ORACLE_HOME/wcportal/es/installES.properties` file and specify the required values. For cluster setup, the following property values should be specified and values should be same for each node of the cluster:

Table 10-2 Property Values

Property	Value
ELASTIC_SEARCH_CLUSTER_NAME	<a unique value to identify the cluster>
ELASTIC_SEARCH_CLUSTER_HOST_LIST	["host1.example.com", "host2.example.com", "host3.example.com"]
ELASTIC_SEARCH_CLUSTER_NODE_LIST	["node-1", "node-2", "node-3"]
ELASTIC_SEARCH_CERTIFICATE_NAME	<Name of the certificate>
ELASTIC_SEARCH_PRIMARY_HOST	<Fully qualified name of the primary host>

where, `host[n].example.com` is the fully-qualified hostname and `node[n]` is the value of `ELASTIC_SEARCH_NODE_NAME` corresponding to `host[n].example.com`.

2. Enter the following command to install Elasticsearch and plugins:

```
ORACLE_HOME/oracle_common/common/bin/wlst.sh
ORACLE_HOME/wcportal/es/installES.py
ORACLE_HOME/wcportal/es/installES.properties
```

The following is the sample output for the successful installation on Linux:

```
...

-> Installing file:////tmp/webcenter-search-es-content-crawl-plugin.zip
-> Downloading file:////tmp/webcenter-search-es-content-crawl-plugin.zip
[================================================] 100%
-> Installed orcl-wcp-es-content-crawl-plugin
-> Please restart Elasticsearch to activate any plugins installed
Generating node certificate for secured communication...
Node certificate generated and password added to elasticsearch keystore
successfully.
Creating the user crawladmin in Elasticsearch server...

...

The user crawladmin is created successfully.
Executing /home/Oracle/product/esHome/startElasticsearch.sh
```

```
Elasticsearch server installation is complete. Elasticsearch server home is at: /
home/Oracle/product/esHome
```

3. If Elasticsearch is being installed on Windows, execute the following command to secure the Elasticsearch server:

```
ORACLE_HOME/wcportal/es/secureESNode.cmd
ORACLE_HOME/wcportal/es/installES.properties
```

4. After the successful installation, the Elasticsearch server will be up and running. Verify that the Elasticsearch server is configured properly.

Install Elasticsearch on Non-Primary Host

Once the installation is successful on the primary host of the cluster, the Elasticsearch server can be installed on other hosts of the cluster (that is non-primary hosts). The procedure to install Elasticsearch server on the non-primary host is as follows:

1. Edit the `ORACLE_HOME/wcportal/es/installES.properties` file and specify the required values. For cluster setup, the following property values should be specified and values should be same for each node of the cluster:

Table 10-3 Property Values

Property	Value
ELASTIC_SEARCH_CLUSTER_NAME	<a unique value to identify the cluster>
ELASTIC_SEARCH_CLUSTER_HOST_LIST	["host1.example.com", "host2.example.com", "host3.example.com"]
ELASTIC_SEARCH_CLUSTER_NODE_LIST	["node-1", "node-2", "node-3"]
ELASTIC_SEARCH_CERTIFICATE_NAME	<Name of the certificate>
ELASTIC_SEARCH_PRIMARY_HOST	<Fully qualified name of the primary host>

2. Enter the following command to install Elasticsearch and plugins:

```
ORACLE_HOME/oracle_common/common/bin/wlst.sh
ORACLE_HOME/wcportal/es/installES.py
ORACLE_HOME/wcportal/es/installES.properties
```

After the successful installation, the Elasticsearch server is not started. The following is the sample output for the successful installation:

```
...

-> Installing file:////tmp/webcenter-search-es-content-crawl-plugin.zip
-> Downloading file:////tmp/webcenter-search-es-content-crawl-plugin.zip
[================================================] 100%
-> Installed orcl-wcp-es-content-crawl-plugin
-> Please restart Elasticsearch to activate any plugins installed

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
++++++++++++++++++++++++++++++++
NOTE: Elasticsearch server installation is not complete. Please copy the required
files from the primary host as mentioned in the documentation and start the server
to complete the installation.
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
++++++++++++++++++++++++++++++++
```

3. If Elasticsearch is being installed on Windows, execute the following command to secure the Elasticsearch server:

   ```
   ORACLE_HOME/wcportal/es/secureESNode.cmd
   ORACLE_HOME/wcportal/es/installES.properties
   ```

4. Copy the node certificate (that is the file specified for `ELASTIC_SEARCH_CERTIFICATE_NAME` property) and the `elasticsearch.keystore` file from the `ES_HOME/esNode/config` folder of the primary host to `ES_HOME/esNode/config` folder of the non-primary host.

5. Start the Elasticsearch server by executing the following command:

   ```
   # applicable if installing on Linux
   ES_HOME/startElasticsearch.sh

   # applicable if installing on Windows
   ES_HOME/startElasticsearch.cmd
   ```

6. Once the Elasticsearch server is up and running, verify that the Elasticsearch server is configured properly.

Validating Cluster Set Up

To validate the cluster set up, access the URL `http://host1.example.com:9200` using the Mozilla Firefox browser.

A basic authentication dialog is displayed to enter the user name and password. Enter your crawl user name and password that you created in WebCenter Portal. After successful login, access the URL `http://host1.example.com:9200/_cluster/health`. Check the response received and the number of nodes should be as shown below:

```
"number_of_nodes": 3
```

This indicates that cluster is set up successfully consisting of three nodes.

> **Note:**
>
> The above steps can be validated using any hostname which is part of the cluster.

Setting Up Load Balancing Using OHS

We can use Oracle HTTP Server (OHS) to do the load balancing in the Elasticsearch cluster. In order to do this, we need to make use of virtual host configuration of OHS. The virtual hosts in OHS can be configured either based on the server name or port number. The sample configuration for port based virtual host is as follows:

1. Create a file `wcp-search.conf` in the `moduleconf` directory (this should be in the directory that has the file `httpd.conf`). The sample configuration for port based virtual host is as follows:

```
<Location />
  WLSRequest ON
  SetHandler weblogic-handler
  WebLogicCluster
host1.example.com:9200,host2.example.com:9200,host3.example.com:9200
  WLIOTimeoutSecs 30000
  WLSocketTimeoutSecs 600
</Location>
```

   When we access OHS server using the configured port, the request will be redirected to one of the hosts mentioned in the WebLogicCluster configuration.

2. Restart the OHS.

Validating Load Balancing Set Up

To validate the load balancing set up, access the OHS URL configured above using the Mozilla Firefox browser.

A basic authentication dialog is displayed to enter the user name and password. Enter your crawl user name and password that you created in WebCenter Portal. After successful login, access the URL `http://OHS-HOST:OHS-PORT/_cluster/health`. Check the response received and the number of nodes should be as shown below:

```
"number_of_nodes": 3
```

This indicates that load balancing using OHS is set up successfully for three nodes.

Securing Elasticsearch Using HTTPS

The Elasticsearch 7.x server can be secured using HTTPS so that the communication between the client and the Elasticsearch server happens over SSL. The following steps are required to configure:

1. Stop the Elasticsearch server.
2. Set the following environment variables:
   - `ES_HOME` - Set this to Elasticsearch Home directory.
   - `JAVA_HOME` - Set this to the Java Home directory.
   - `PATH` - Update to use `$JAVA_HOME/bin`.
3. On any single node, generate a certificate and private key for the nodes in the cluster. When prompted, provide a password for the certificate.

```
$ES_HOME/elasticsearch/bin/elasticsearch-certutil cert -s -out $ES_HOME/
esNode/config/https-certificate.p12 --dns <DNS_name>
```

   Where `<DNS_name>` is the comma-separated list of fully qualified hostnames of the nodes of the cluster.

4. The password used above should be stored in the Elasticsearch Keystore, run the following commands to store the password:

```
$ES_HOME/elasticsearch/bin/elasticsearch-keystore add
xpack.security.http.ssl.keystore.secure_password
$ES_HOME/elasticsearch/bin/elasticsearch-keystore add
xpack.security.http.ssl.truststore.secure_password
```

5. Copy `$ES_HOME/esNode/config/https-certificate.p12` and `$ES_HOME/esNode/config/elasticsearch.keystore` files to `$ES_HOME/esNode/config` directory of every node in the cluster.
6. On each node of the cluster, add the following configuration to the `$ES_HOME/esNode/config/elasticsearch.yml` file:

```
xpack.security.http.ssl.enabled: true
xpack.security.http.ssl.verification_mode: certificate
xpack.security.http.ssl.keystore.path: https-certificate.p12
xpack.security.http.ssl.truststore.path: https-certificate.p12
```

7. Start the Elasticsearch server.

Adding Certificate to WebCenter Portal Keystore

The Elasticsearch server certificate must be added to WebCenter Portal Keystore to establish the trust between the client and server.

Download Certificate from Elasticsearch Server

The following steps can be used to obtain the certificate from the Elasticsearch server:

1. Open the Firefox browser and connect to the Elasticsearch server with the following command:

```
https://host_name:9200
```

   where `host_name` is the name of the Elasticsearch server.
2. Accept the security exception and continue. Provide the login credentials when prompted.
3. Click the **Lock** icon in the **URL** field and navigate to **Connection not secure** and then **More Information**.
4. In the popup window, click the **View Certificate** button.
5. Click on the link **PEM (cert)** to download the certificate in the *.PEM* format.

Add the Certificate to WebCenter Portal Keystore

Once the certificate is downloaded, it should be imported to WebCenter Portal Keystore.

To import:

1. Execute the following command in WebCenter Portal server and enter the Keystore password when prompted:

```
keytool -importcert -alias es_cert -file /filepath/sslcertificate/
es_certificate.pem -keystore <JDK_path>/jre/lib/security/cacerts
```

2. Restart the Managed server.

Search Connection to Elasticsearch Server Running on HTTPS

If the search connection doesn't exist, the search connection to the Elasticsearch server running on HTTPS can be created using the following command:

```
createSearchConnection(appName='webcenter', name='<conn_name>', url='https://
<ES_host>:<ES_port>', indexAliasName='<index_alias>', appUser='<app_user>',
appPassword='<app_user_password>')
```

If the search connection already exists, it should be updated to use HTTPS and can be done using the following command:

```
setSearchConnection(appName='webcenter', name='<existing_conn_name>',
indexAliasName='<existing_index_alias>', appUser='<app_user>',
appPassword='<app_user_password>', useSSL=1)
```

Integrating with OCI Search Service with OpenSearch

Configure OpenSearch to index and search objects in WebCenter Portal.

Topics:

- Understanding OCI Search Service with OpenSearch
- Prerequisites for Configuring OCI Search Service with OpenSearch
- Configuration Roadmap for OCI Search Service with OpenSearch in WebCenter Portal
- Create a OpenSearch Instance in OCI
- Import OpenSearch SSL Certificate into WebCenter Portal Keystore

Understanding OCI Search Service with OpenSearch

Oracle Cloud Infrastructure (OCI) Search Service with OpenSearch is an insight engine offered as an Oracle-managed service. Without any downtime, Oracle automates patching, updating, upgrading, backing up, and resizing the service. You can store, search, and analyze large volumes of data quickly and see results in near real-time.

OCI Search Service combines proven OpenSearch technology with the flexibility of OCI.

You can configure OCI Search Service with OpenSearch to search the following resources in WebCenter Portal:

- Documents, including wikis and blogs.
- Portals, page metadata, page content (contents of HTML, Text, and Styled Text components), lists, and people resources.

- Announcements and discussions (available only for portals upgraded from prior releases).

> **Note:**
>
> Oracle WebCenter Portal can be hosted and run on-premise, but it is always recommended to host WebCenter Portal on OCI for leveraging the integration with OCI OpenSearch in the most optimum way while running WebCenter Portal on Cloud.

Prerequisites for Configuring OCI Search Service with OpenSearch

Ensure the following requirements:

1. Get OCI tenancy details.
2. Create the required service policies in the Oracle Cloud Console, tailoring them to your needs. For example, change any user to the desired group, and provide the path to the compartment if required.
   Working policies:

   ```
   Allow group <your_group> to manage opensearch-family in compartment
   opensearch
   Allow service opensearch to manage vcns in compartment opensearch
   Allow service opensearch to manage vnics in compartment opensearch
   Allow service opensearch to use subnets in compartment opensearch
   Allow service opensearch to use network-security-groups in
   compartment opensearch
   ```

3. Create a VCN with a public subnet and a private subnet.
   - Simplified process:
     a. Open the Oracle Cloud Console navigation menu.
     b. Click **Networking**. and then click **Virtual Cloud Networks**.
     c. Click **Start VCN Wizard**, and then click **Create VCN with Internet Connectivity**.
   - Custom process:
     a. Open the Oracle Cloud Console navigation menu.
     b. Click **Networking**. and then click **Virtual Cloud Networks**.
     c. Click **Create VCN** and provide your own desired details.
4. Create a VM instance in the public subnet of the VCN.
   a. Open the Oracle Cloud Console navigation menu.
   b. Click **Compute**, and then click **Instances**.
   c. Click **Create instance**.
   d. Enter the **Name**, and select the compartment.
   e. In the **Image and shape** area, use the default values (Oracle Linux 7, VM.Standard.E4.Flex).

f. In the **Networking** area, select the public subnet and assign a public IP.

g. In the **Add SSH keys** area, decide whether you want to use an existing SSH key, or generate a new SSH key. If you choose to generate a new SSH key, remember to download the key.

Configuration Roadmap for OCI Search Service with OpenSearch in WebCenter Portal

Table 10-4 Roadmap - Setting Up OCI Search Service with OpenSearch in WebCenter Portal

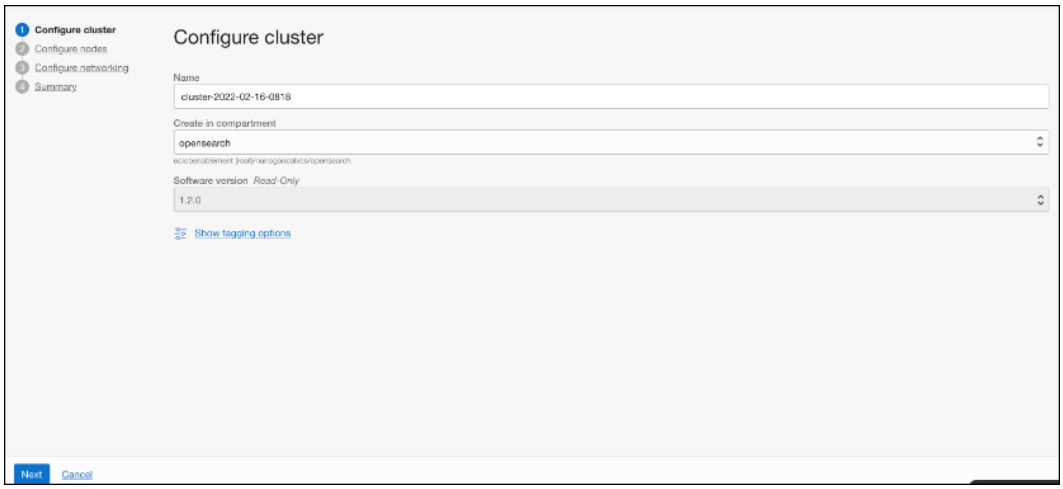
Actor	Task
Administrator	Creating a Crawl Admin User in WebCenter Portal
Administrator	Create a OpenSearch Instance in OCI
Administrator	Import OpenSearch SSL Certificate into WebCenter Portal Keystore
Administrator	Configuring WebCenter Content for Search
Administrator	Configuring WebCenter Portal for Search
Administrator	Configuring Search Crawlers
Administrator	(Optional) Customizing Search Settings in WebCenter Portal Administration
Administrator	(Optional) Modifying Search Global Attributes
Administrator	(Optional) Configuring Search Custom Attributes
Administrator	(Optional) Creating Custom Facets

Create a OpenSearch Instance in OCI

Task 1: Create an OCI Search Service cluster

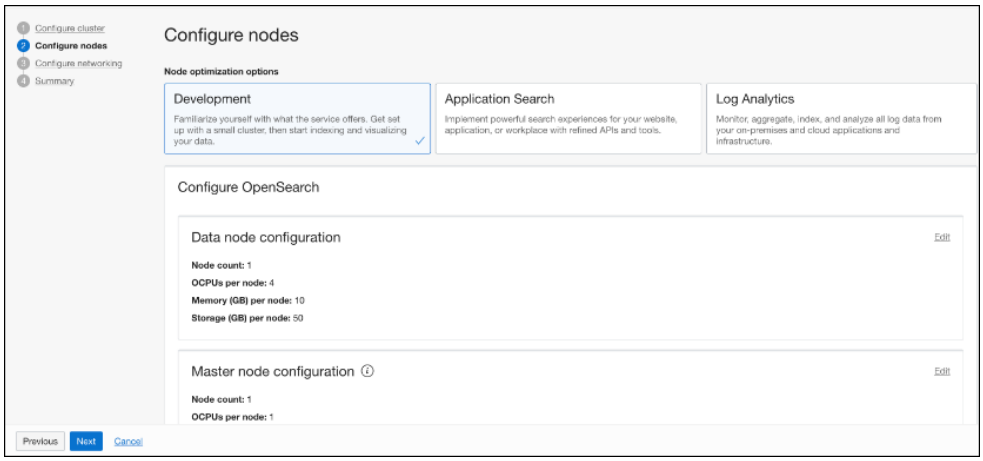
1. Open the Oracle Cloud Console navigation menu. Click **Databases**, **OCI Search Service**, and then click **Clusters**. Then, click **Create cluster**. Choose the cluster name and compartment where you want to create the cluster.

Figure 10-1 Create Cluster



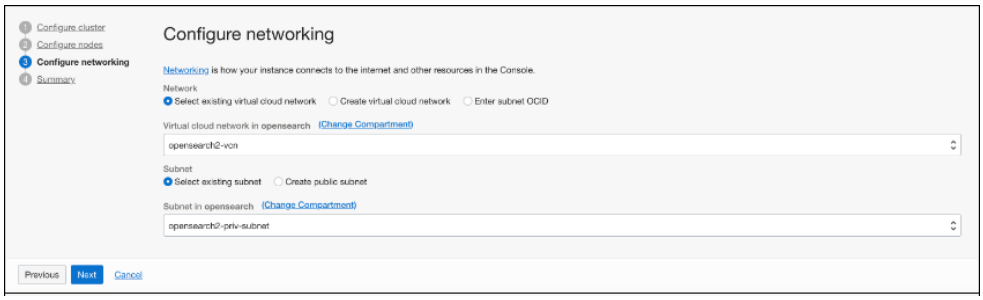
2. Click **Next**.
3. Choose the cluster sizing, and then click **Next**.

Figure 10-2 Configure Nodes



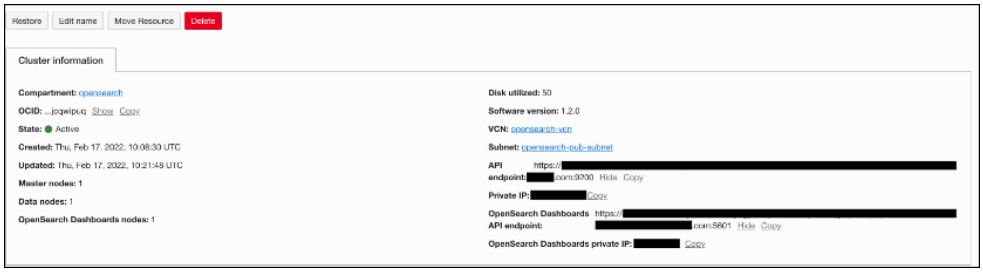
4. Select the VCN you created and then select the private subnet.

Figure 10-3 Configure Networking



5. Click **Next**. After the cluster creation, in the OCI Search Service cluster details page, note the API endpoints and the IP addresses which you can alternatively use.

Figure 10-4 Cluster Information



Task 2: Create security rules in the VCN Security List

In the VCN, create a Security List with the following security rules. Alternatively, they can be added to the VCN Default Security List.

1. Open the VCN details page, click **Security Lists**, select the **Security List**, and then click **Add Ingress Rule**.

> **Note:**
>
> Add a rule for port 9200 (OpenSearch), and a rule for 5601 (OpenSearch Dashboards).

Figure 10-5 TCP traffic for ports: 9200

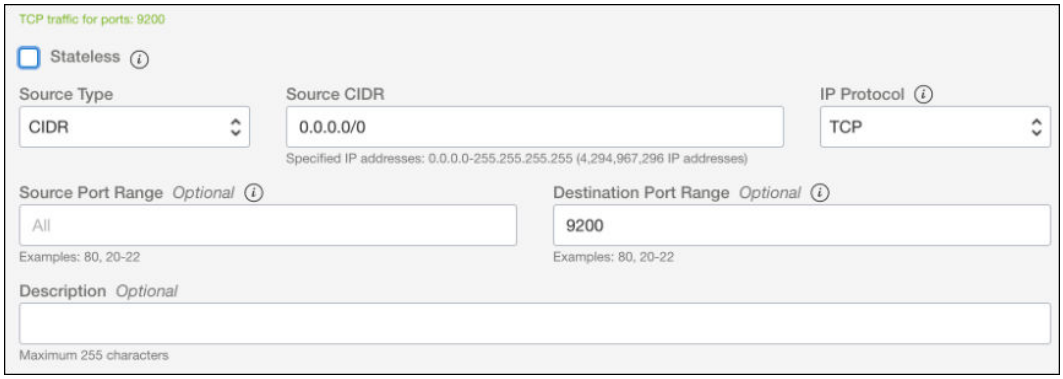


Figure 10-6 TCP traffic for ports: 5601

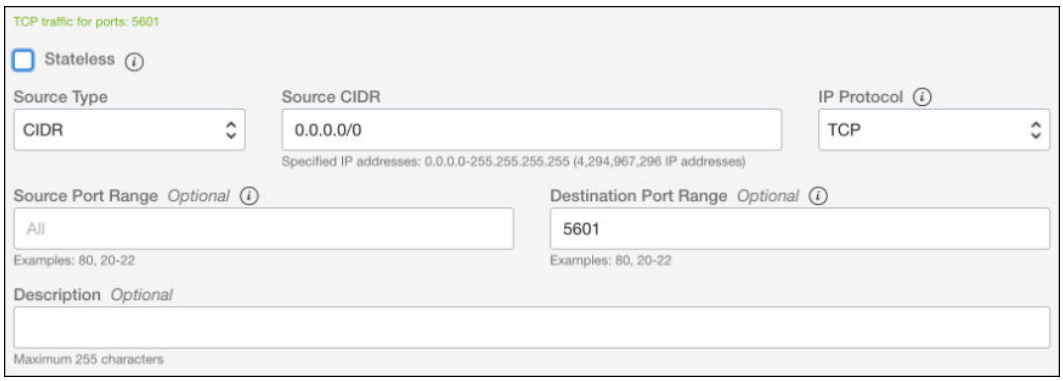
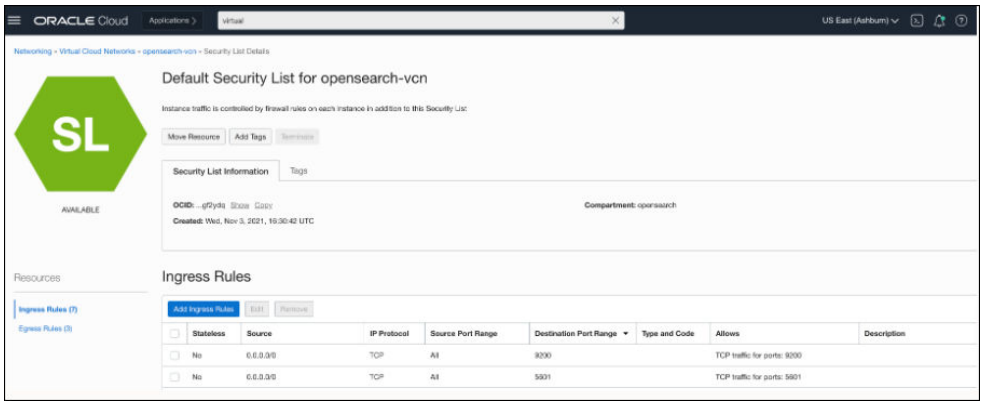


Figure 10-7 Security List



Task 3: Test the connection to OCI Search Service – OpenSearch endpoint

Use one of the following options:

1. **Option 1**: From inside the created VM instance
   a. Connect to the instance via SSH:

   ```
   ssh -i ~/.ssh/id_rsa_opensearch.key
   opc@<your_VM_instance_public_IP>
   ```

   b. Run one of the following commands:

   ```
   curl https://mycluster.opensearch.us.example.com:9200
   # OpenSearch API endpoint example
   ```

   ```
   curl https://<your_opensearch_private_IP>:9200 --insecure
   # OpenSearch private IP example
   ```

2. **Option 2**: From your local machine, through port forwarding
   a. Run the following port forwarding SSH command in the Terminal. Do not close the Terminal to ensure that the connection remains active.

   ```
   ssh -C -v -t -L
   127.0.0.1:5601:<your_opensearch_dashboards_private_IP>:5601 -L
   127.0.0.1:9200:<your_opensearch_private_IP>:9200
   opc@<your_VM_instance_public_IP> -i <path_to_your_private_key>
   ```

   b. Open a new Terminal window and run the following command:

   ```
   curl https://localhost:9200 --insecure
   ```

   If the steps are performed correctly you should see a response as follows, regardless of the option used:

   ```
   {
   "name" : "opensearch-master-0",
   "cluster_name" : "opensearch",
   ```

```
"cluster_uuid" : "M6gclrE3QLGEBlkdme8JkQ",
"version" : {
   "distribution" : "opensearch",
   "number" : "1.2.4-SNAPSHOT",
   "build_type" : "tar",
   "build_hash" : "e505b10357c03ae8d26d675172402f2f2144ef0f",
   "build_date" : "2022-02-08T16:44:39.596468Z",
   "build_snapshot" : true,
   "lucene_version" : "8.10.1",
   "minimum_wire_compatibility_version" : "6.8.0",
   "minimum_index_compatibility_version" : "6.0.0-beta1"
},
"tagline" : "The OpenSearch Project: https://opensearch.org/"
}
```

Task 4: Note down the Private IP of the OpenSearch cluster

You can note down the Private IP of the cluster as shown in the figure below.

Figure 10-8 Cluster Information

Import OpenSearch SSL Certificate into WebCenter Portal Keystore

To import OpenSearch SSL certificate:

1. Log in to the public VM created in public subnet (see Step 2 of Prerequisites for Configuring OCI Search Service with OpenSearch).

   ```
   ssh -i ~/.ssh/id_rsa_opensearch.key opc@<your_VM_instance_public_IP>
   ```

2. Run the following command in the public VM to get the SSL certificate of OCI OpenSearch:

   ```
   $ echo -n | openssl s_client -showcerts -connect
   Opensearch_Private_IP:9200 -servername opensearch.us-
   ashburn-1.oci.oracleiaas.com|sed -ne '/-BEGIN CERTIFICATE-/,/-END
   CERTIFICATE-/p' > /tmp/opensearch_cert_chain.crt
   ```

   Where `servername` is the name of server on which OpenSearch cluster is running , servername is the last part of API endpoint excluding OCI ID. (refer to the image above)

3. For WebCenter Portal to trust OpenSearch certificate, the OpenSearch certificate must be imported into the WebCenter Portal trust store. Import the certificate in WebCenter Portal trust store using the following WLST commands:

```
$ svc=getOpssService(name='KeyStoreService')
$
svc.importKeyStoreCertificate(appStripe='system',name='trust',passwo
rd='',alias='opensearch_cert_chain',type='TrustedCertificate',filepa
th='/tmp/opensearch_cert_chain.crt',keypassword='')
$ syncKeyStores(appStripe='system',keystoreFormat='KSS')
$ exit()
```

4. To disable hostname verification:
   a. Log in to the WebLogic console.
   b. Click the **Lock & Edit** button.
   c. Select **Servers** from the left navigation, select **Sever Name**, then **Configuration**, then **SSL**, and **Advanced**. Select **None** from the **Hostname Verification** drop-down menu. Click **Save** and activate the changes.

Creating a Crawl Admin User in WebCenter Portal

You can designate an existing user as crawl admin or create a crawl admin user (for example, `mycrawladmin`) in WebCenter Portal and in your back-end identity management server to search using Elasticsearch. You must create a crawl admin user only once.

> **Note:**
>
> See your identity management system documentation for information on creating users.

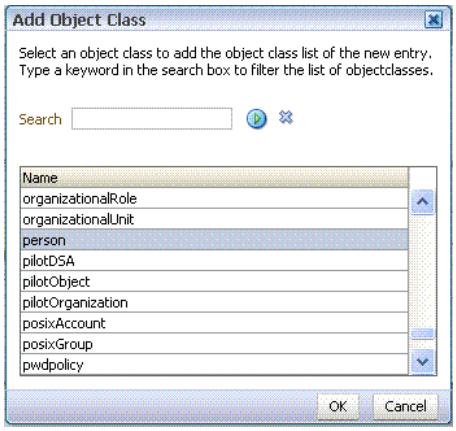
The following example uses Oracle Directory Services Manager to create the `mycrawladmin` user:

1. 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"`.

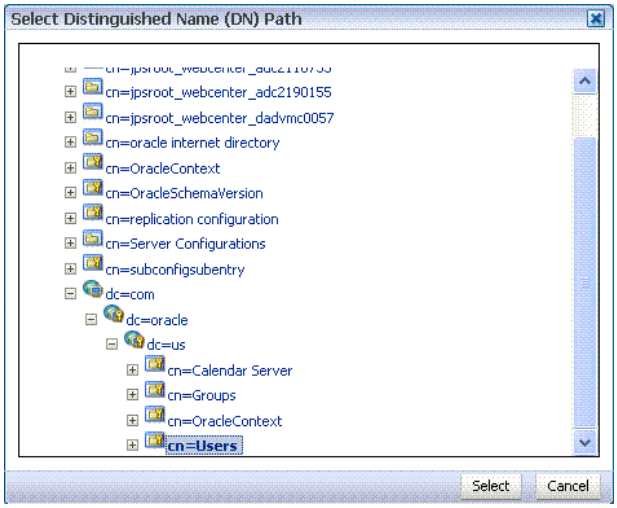
2. In the Add Object Class dialog, select the appropriate object class, and click **OK**.

Figure 10-9 Oracle Directory Services Manager - Add Object Class



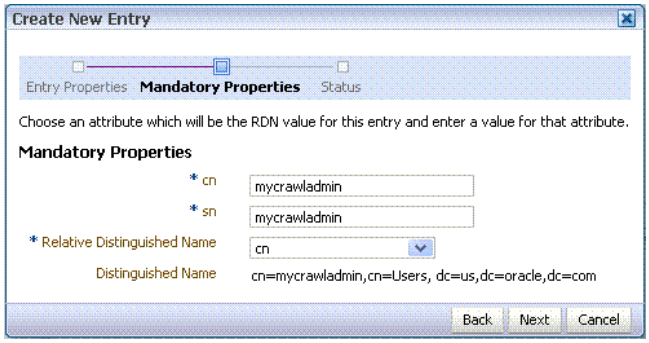
3. Find the distinguished name (DN) path, and click **Select**.

   This example selects `"dc=com,dc=oracle,dc=us,cn=Users"`.



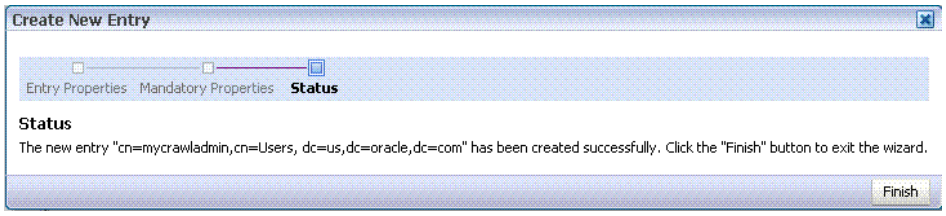
4. In the Create New Entry dialog, enter properties, and click **Next**.

Figure 10-10 Oracle Directory Services Manager - Create New Entry



5. When you see that the new entry was created successfully, click **Finish**.

Figure 10-11 Oracle Directory Services Manager - Status



Configuring WebCenter Content for Search

This topic describes how to configure WebCenter Content for search.

> **Note:**
>
> The following topics are applicable only if WebCenter Content is configured.

- Creating a Crawl User in WebCenter Content
- Configuring the `SESCrawlerExport` Component

Creating a Crawl User in WebCenter Content

This procedure describes how to create a new crawl user in WebCenter Content.

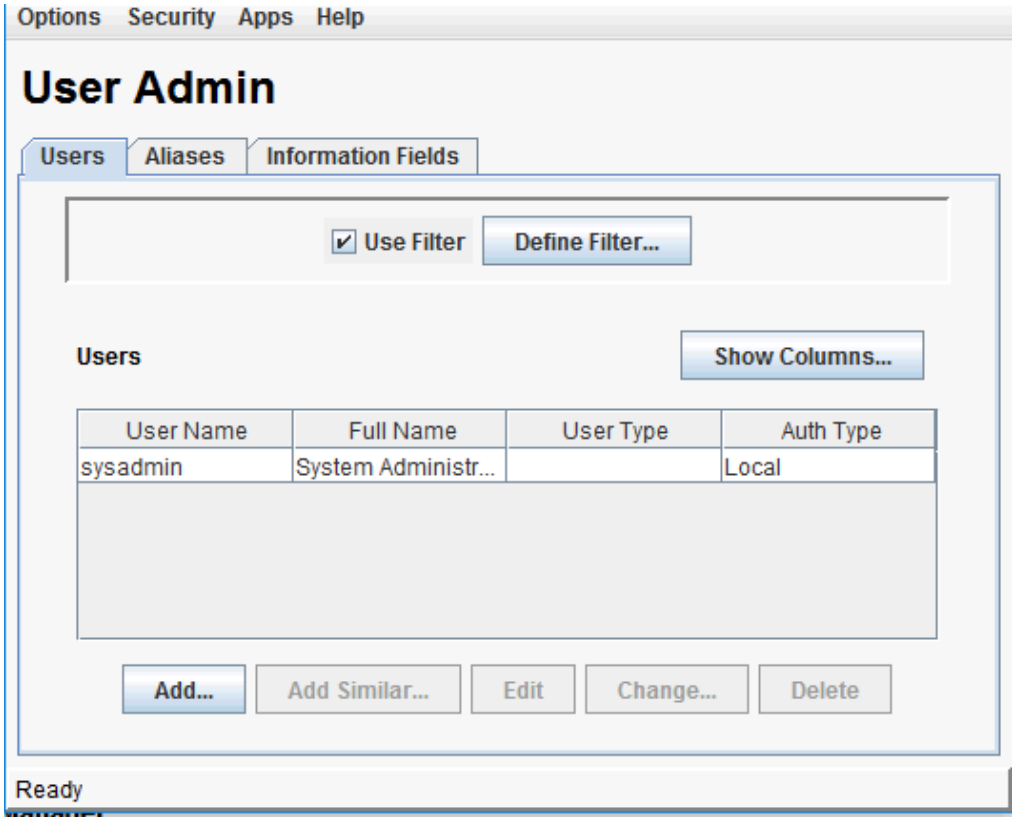
If you want users with the admin role to crawl, then use an admin user account as the crawl user. If you want non-admin users to crawl, then create a new crawl user.

To create a crawl user in WebCenter Content:

1. Log on to WebCenter Content as an Administrator.
2. To create a role `sescrawlerrole`, do the following:
   a. On the WebCenter Content home page, select **Admin Applets** from the **Administration** menu.
   b. Select **User Admin**.

      The **User Admin** dialog opens.

Figure 10-12 User Admin Page



c. In the User Admin dialog, select **Security** tab, then **Permissions by Role**.

d. In the Permissions By Role dialog, click **Add New Role**.

e. In the Add New Role dialog, specify **Role Name** and **Role Display Name** as `sescrawlerrole`.

f. Click **OK**.

3. To create a user `sescrawler`, and assign the `sescrawlerrole` role to the user, do the following:

a. On the WebCenter Content home page, select **Admin Applets** from the **Administration** menu.

b. Select **User Admin**.

The **User Admin** dialog opens.

c. On the **Users** tab, click **Add**.

d. Choose the Authentication Type as *Local* and click **OK**.

e. In the Add User dialog, enter the name as `sescrawler` and specify password.

f. On the Roles tab, click **Add Role** and select `sescrawlerrole` to assign the role to the user.

g. Click **OK**.

h. Close the **User Admin** dialog.

4. On the WebCenter Content home page, expand **Administration**, then **Admin Server**. Select **General Configuration** and append the `sceCrawlerRole=sescrawlerrole` entry in the **Additional Configuration Variables** section.
5. Restart WebCenter Content.

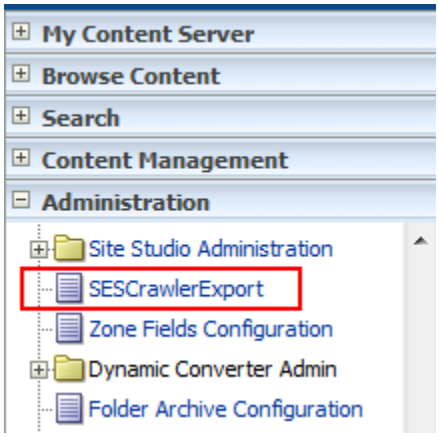
Configuring the `SESCrawlerExport` Component

Before you begin, verify that the `SESCrawlerExport` component is enabled. If not, enable the component (see Enabling the WebCenterConfigure Component) and restart the WebCenter Content server.

To configure the `SESCrawlerExport`component for admin and non-admin users:

1. Log on to WebCenter Content as a system administrator.
2. On the WebCenter Content home page, expand **Administration** and select **SESCrawlerExport**.

Figure 10-13 WebCenter Content Server Administration Menu



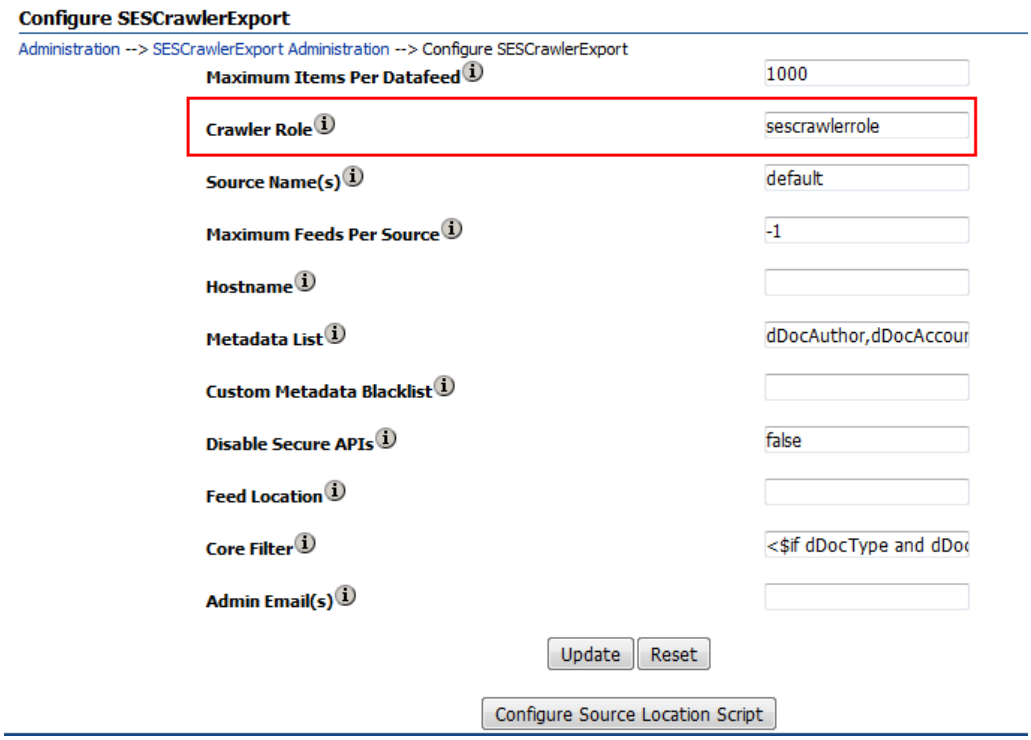
3. On the SESCrawlerExport Administration page, click **Configure SESCrawlerExport**.

Figure 10-14 SESCrawlerExport Administration Page

4. On the Configure SESCrawlerExport page, set the value for **Crawler Role**.
   - If you want to configure SESCrawlerExport for an admin user, set the value for the **Crawler Role** as `admin`.
   - If you want to configure SESCrawlerExport for a non-admin user, set the value for the **Crawler Role** as `sescrawlerrole`.

The following figure shows the Configure SESCrawlerExport page, where **Crawler Role** is set as `sescrawlerrole` for a non-admin user.

Figure 10-15 Configure SESCrawlerExport Page



5. Click **Update**.
6. Restart WebCenter Content.

Configuring WebCenter Portal for Search

To configure WebCenter Portal for search, you need to configure the connection between WebCenter Portal and Elasticsearch or OCI Search Service with OpenSearch.

Topics:

- Configuring WebCenter Portal for Elasticsearch
- Configuring WebCenter Portal for OCI Search Service with OpenSearch

Configuring WebCenter Portal for Elasticsearch

To configure WebCenter Portal for search, you need to configure the connection between WebCenter Portal and Elasticsearch and you have to configure the WebCenter Content crawl user and WebCenter Content admin user in Elasticsearch.

1. Navigate to your Oracle home directory and invoke the WLST script.

   See Running Oracle WebLogic Scripting Tool (WLST) Commands.

2. Connect to the Oracle WebCenter Portal domain (`WC_Portal`) server.

3. At the WLST command prompt, run the `createCred` WLST command to configure the WebCenter Content crawl user in Elasticsearch.

   ```
   createCred(map="oracle.es.security", key="content.crawl.credentials",
   user='wcc-crawl-user', password='wcc-crawl-password', desc="UCM Crawl
   User")
   ```

   where,

   - `wcc-crawl-user` is the WebCenter Content crawl user. See Creating a Crawl User in WebCenter Content.
   - `wcc-crawl-password` is the password of the WebCenter Content crawl user.
   - `desc` is the description of the WebCenter Content crawl user.

   The following example shows how to configure the WebCenter Content crawl user:

   ```
   createCred(map="oracle.es.security", key="content.crawl.credentials",
   user='sescrawler', password='welcome1', desc="UCM Crawl User")
   ```

4. At the WLST command prompt, run the `createCred` WLST command to configure the WebCenter Content admin user in Elasticsearch.

   ```
   createCred(map="oracle.es.security", key="content.admin.credentials",
   user='wcc-admin-user', password='wcc-admin-password', desc="UCM Admin
   User")
   ```

   where,

   - `wcc-admin-user` is the WebCenter Content Admin user.
   - `wcc-admin-password` is the WebCenter Content Admin password.
   - `desc` is the description of the WebCenter Content Admin user.

The following example shows how to configure the WebCenter Content admin user:

```
createCred(map="oracle.es.security", key="content.admin.credentials",
user='weblogic', password='welcome1', desc="UCM Admin User")
```

5. To confirm that the user credentials are added:
   - Connect to the Fusion Middleware Control, `http:<host:port>/em`.
   - From the **WebLogic Domain** drop-down menu, select **Security > Credentials**.
   - Expand **oracle.es.security**.
   - Ensure you have two credentials:
     - content.admin.credentials: For the WebCenter Content admin user for Elasticsearch (for example, weblogic or any other user with administration permissions in the Content server)
     - content.crawl.credentials: For the WebCenter Content crawl user for Elasticsearch (for example, sescrawler user or the user you created for doing Elasticsearch crawling)
6. Stop and start the Elasticsearch server.
7. At the WLST command prompt, run the `createSearchConnection` WLST command to configure a connection between WebCenter Portal and Elasticsearch:

   ```
   createSearchConnection(appName, name, url, indexAliasName, appUser,
   appPassword)
   ```

   where
   - `appName` is the name of the application, for WebCenter Portal, the value is `webcenter`.
   - `name` is the connection name. The name must be unique within the application. For example `dev-es`.
   - `url` is the location of the Elasticsearch server. For example, `http://host:9200`

     Note: If you choose to connect to an Elasticsearch cluster. Enter the URL of the load balancer, for example `https://OHS-host:port`
   - `indexAliasName` is the name of the index alias in the Elasticsearch server. For example, `webcenter_portal`.

     The indices will be created using the alias as the prefix, as shown in the following example:
     - `<indexAliasName>_portals` For example, `webcenter_portal_portals`
     - `<indexAliasName>_documents` For example, `webcenter_portal_documents`

     Note: The name must be in lowercase alphanumeric characters and unique across all portal servers.
   - `appUser` is the crawl admin user name. For example, `mycrawladmin`.
   - `appPassword` is the crawl admin user password.

   The following example creates a connection between WebCenter Portal (`webcenter`) and Elasticsearch located at `http://host:9200`:

```
createSearchConnection (appName='webcenter',name='dev-es',
url='http://host:9200', indexAliasName='webcenter_portal',
appUser='mycrawladmin', appPassword='welcome1')
```

The following example creates a connection between WebCenter Portal (webcenter) and Elasticsearch cluster located at `http://OHS-HOST:OHS-PORT`:

```
createSearchConnection (appName='webcenter',name='dev-es',
url='http://OHS-HOST:OHS-PORT', indexAliasName='webcenter_portal',
appUser='mycrawladmin', appPassword='welcome1')
```

Configuring WebCenter Portal for OCI Search Service with OpenSearch

To configure WebCenter Portal for search, you need to configure the connection between WebCenter Portal and OCI Search Service with OpenSearch and grant the crawl application role to the crawl admin user.

1. Navigate to your Oracle home directory and invoke the WLST script.

   See Running Oracle WebLogic Scripting Tool (WLST) Commands.

2. Connect to the Oracle WebCenter Portal domain (`WC_Portal`) server.

3. At the WLST command prompt, run the `grantAppRole` WLST command to grant the crawl admin user the application role so that it may perform searches on behalf of WebCenter Portal users.

   ```
   grantAppRole(appStripe='webcenter', appRoleName='webcenter#-
   #defaultcrawl',
   principalClass='weblogic.security.principal.WLSUserImpl',
   principalName='crawl-admin-user')
   ```

   where `crawl-admin-user` is the crawl admin user.

   The following example shows how to grant the crawl admin user the application role:

   ```
   grantAppRole(appStripe='webcenter', appRoleName='webcenter#-
   #defaultcrawl',
   principalClass='weblogic.security.principal.WLSUserImpl',
   principalName='mycrawladmin')
   ```

4. At the WLST command prompt, run the `createSearchConnection` WLST command to configure a connection between WebCenter Portal and Elasticsearch:

   ```
   createSearchConnection(appName, name, url, indexAliasName, appUser,
   appPassword)
   ```

   where

   - `appName` is the name of the application, for WebCenter Portal, the value is `webcenter`.
   - `name` is the connection name. The name must be unique within the application. For example `dev-es`.
   - `url` is the location of the OCI Search Service with OpenSearch server. For example, `https://OpensearchPrivateIP:9200`

- `indexAliasName` is the name of the index alias in the OCI Search Service with OpenSearch server. For example, `webcenter_portal`.

  The indices will be created using the alias as the prefix, as shown in the following example:

  - `<indexAliasName>_portals` For example, `webcenter_portal_portals`
  - `<indexAliasName>_documents` For example, `webcenter_portal_documents`

  Note: The name must be in lowercase alphanumeric characters and unique across all portal servers.

- `appUser` is the crawl admin user name. For example, `mycrawladmin`.
- `appPassword` is the crawl admin user password.

The following example creates a connection between WebCenter Portal (`webcenter`) and OCI Search Service with OpenSearch located at `https://OpensearchPrivateIP:9200`:

```
createSearchConnection (appName='webcenter',name='dev-es', url='https://
OpensearchPrivateIP:9200', indexAliasName='webcenter_portal',
appUser='mycrawladmin', appPassword='welcome1')
```

Synchronizing Users in WebCenter Portal

Before performing a portal full crawl, we recommend you to run the LDAP synchronization WLST command to ensure that all users are available in portal.

To synchronize users in WebCenter Portal:

1. Navigate to your Oracle home directory and invoke the WLST script.

   See Running Oracle WebLogic Scripting Tool (WLST) Commands.

2. Connect to the Oracle WebCenter Portal domain (`WC_Portal`) server.
3. At the WLST command prompt, run the `startSyncProfiles` WLST command to synchronize profile information.

   ```
   startSyncProfiles(appName='webcenter')
   ```

   > **Note:**
   >
   > Synchronizing profile information may take some time depending on the number of users.

4. Run the `isSyncProfilesRunning` command to verify that the user synchronization is complete.

   ```
   isSyncProfilesRunning(appName='webcenter')
   ```

Configuring Search Crawlers

You can configure the following types of crawlers to index WebCenter Portal resources:

- Portal Crawler: This uses the Portal crawl source to crawl certain objects, such as lists, page metadata, page content (contents of HTML, Text, and Styled Text components), portals, and profiles.

- Documents Crawler: This uses the Documents crawl source to crawl documents, including wikis and blogs.
- Discussions Crawler: This uses the Discussions crawl source to crawl discussion forums and announcements. This option is available only for portals upgraded from prior releases that include Discussions.

The following topics describe how to create different crawl sources using Scheduler UI in WebCenter Portal Administration:

- Creating a Portal Crawl Source
- Creating a Documents Crawl Source
- Taking a Snapshot of the Content
- Creating a Discussions Crawl Source

Creating a Portal Crawl Source

To create a crawl source to crawl objects such as lists, page metadata, page content (contents of HTML, Text, and Styled Text components), portals, and profiles:

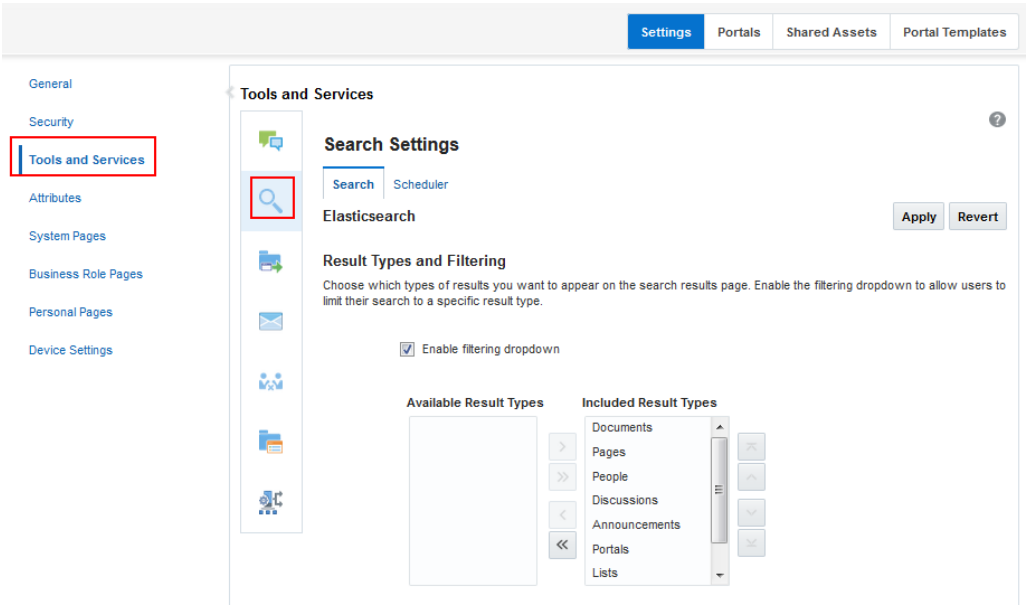
1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Tools and Services**.

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

   `http://host:port/webcenter/portal/admin/settings/tools`

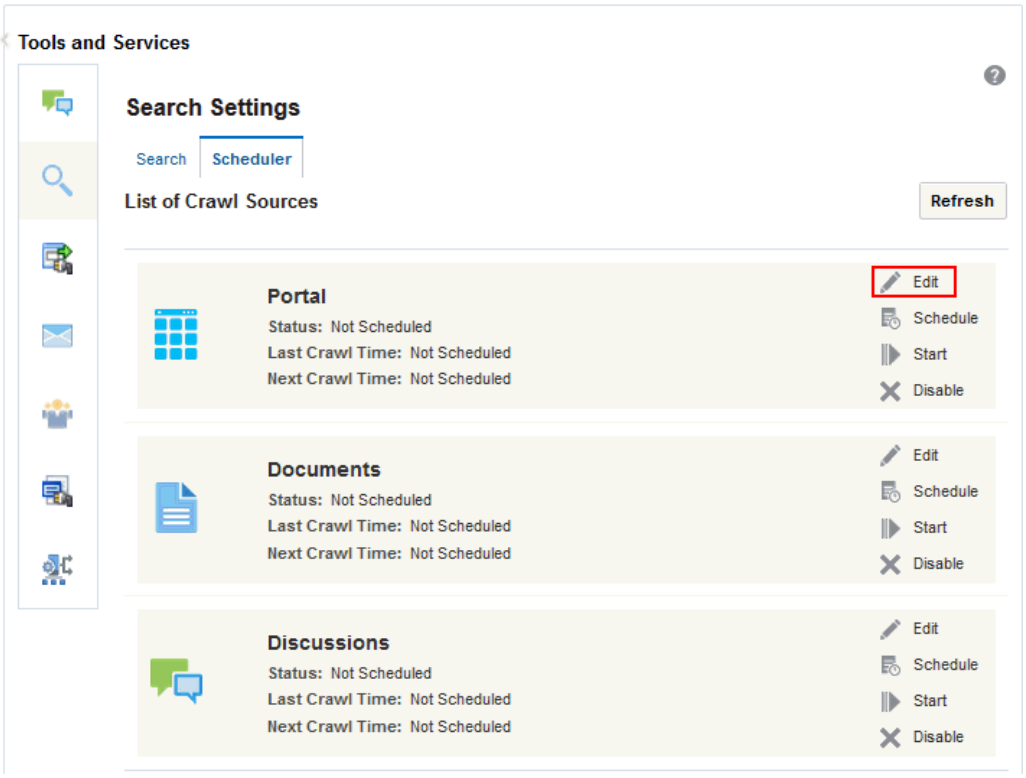
2. Click the icon for Search to open the Search Settings page.

Figure 10-16 Elasticsearch Settings



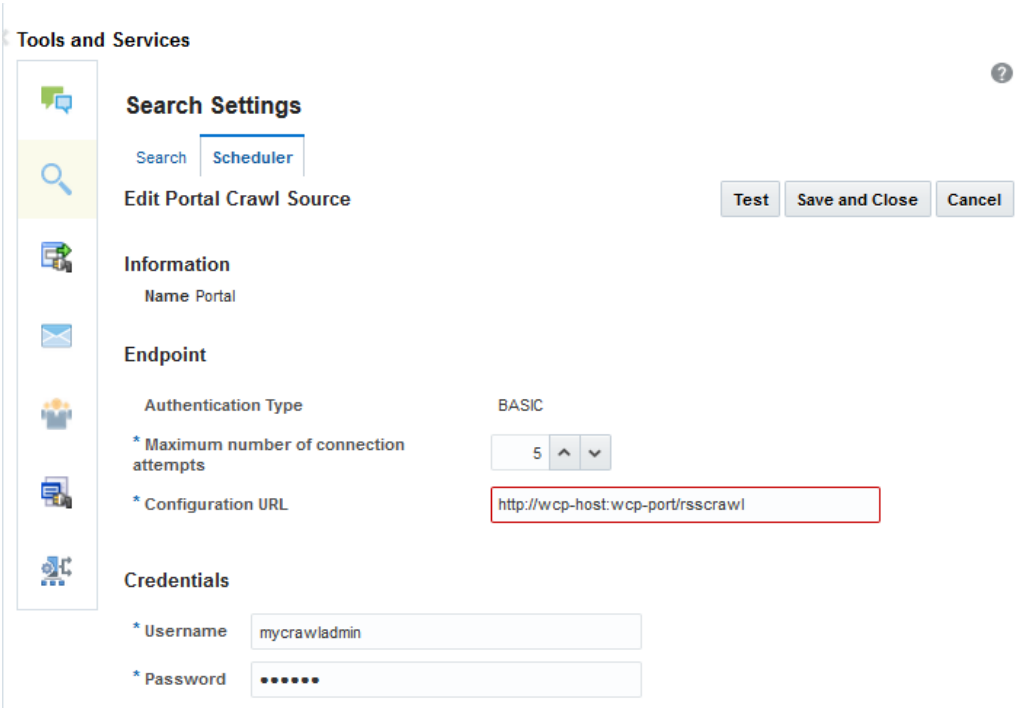
3. On the **Scheduler** tab, select the **Portal** crawl source and click **Edit**.

Figure 10-17 Elastic Search Setting Scheduler



4. On the Edit Portal Crawl Source page, modify the following source parameters as desired:
   - **Maximum number of connection attempts**: Maximum number of connection attempts to access the configuration URL. Choose number from 2 to 10.
   - **Configuration URL**: URL of the RSS crawl servlet. For example: `http://wcp-host:wcp-port/rsscrawl`

Figure 10-18 Elastic Search Edit Portal Crawl Source



5. Enter the WebCenter Portal crawl admin user credentials.
6. Click **Test** to test the connection.
7. Click **Save and Close** to save the changes.

Creating a Documents Crawl Source

If you have configured WebCenter Content, you can create a crawl source for documents, including wikis and blogs:

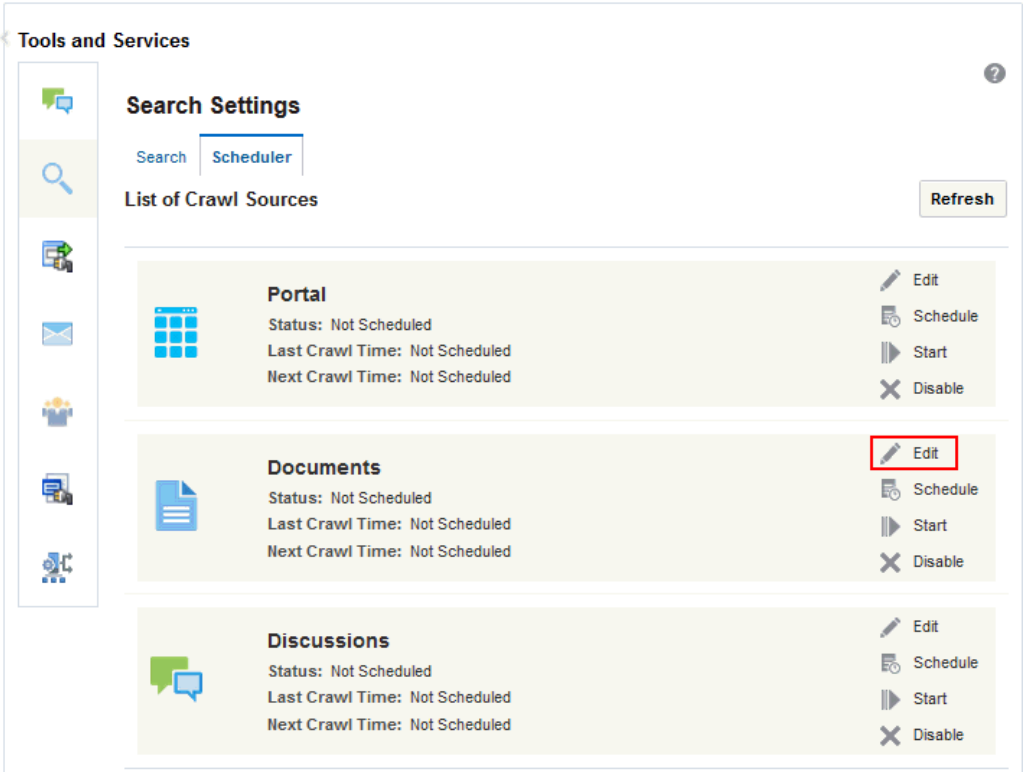
1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Tools and Services**.

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

   `http://host:port/webcenter/portal/admin/settings/tools`

2. Click the icon for Search to open the Search Settings page.
3. On the **Scheduler** tab, select the **Documents** crawl source and click **Edit**.

Figure 10-19 Elastic Search Setting Scheduler



4. On the Edit Document Crawl Source page, modify the following source parameters as desired:

   - **Maximum number of connection attempts**: Maximum number of connection attempts to access the configuration URL. Choose number from 2 to 10.
   - **Configuration URL**: URL of the Webcenter Content SESCrawlerExport component.

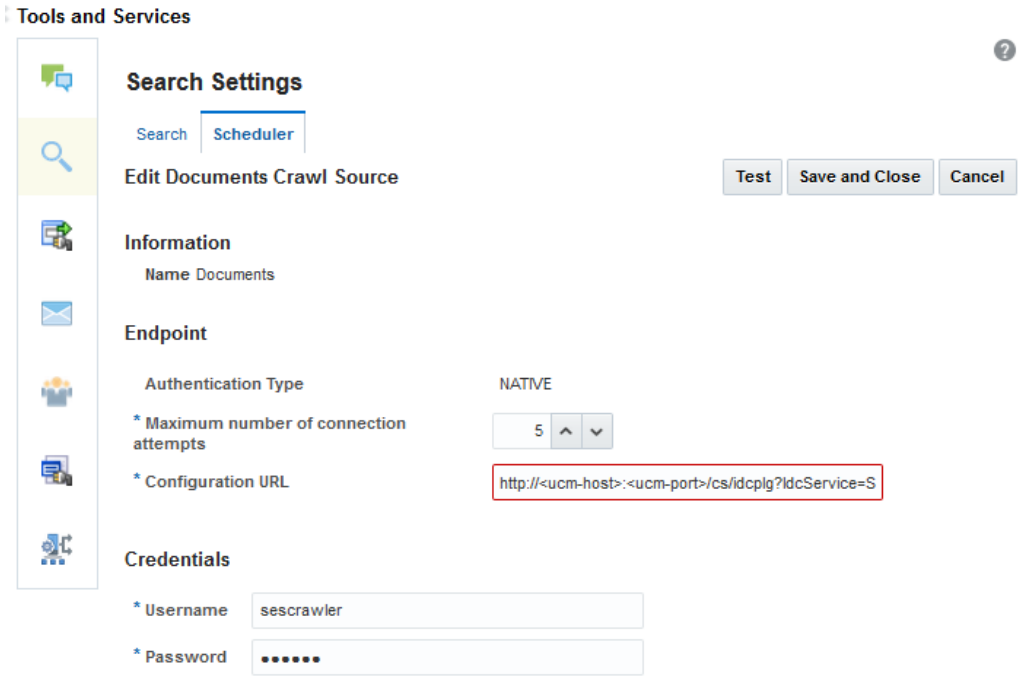
     For example: `http://wcc-host:wcc-port/cs/idcplg?IdcService=SES_CRAWLER_DOWNLOAD_CONFIG&source=source_name`

     The *`source_name`* must be one of the strings used in the WebCenter Content SESCrawlerExport component Source Name (`sceSourceName`) parameter.

     For more information on SESCrawlerExport parameters, see Configuring SESCrawlerExport Parameters in *Administering Oracle WebCenter Content*.

     For example: `http://host.example.con:port/cs/idcplg?IdcService=SES_CRAWLER_DOWNLOAD_CONFIG&source=esDS`

Figure 10-20 Elastic Search Edit Document Crawl Source



5. Enter the WebCenter Content crawl admin user login credentials.
6. Click **Test** to test the connection.
7. Click **Save and Close** to save the changes.

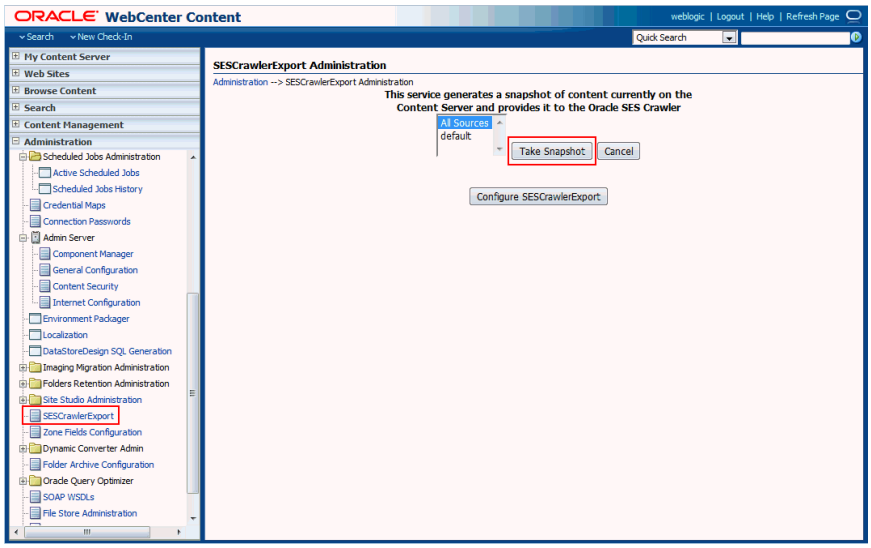
Taking a Snapshot of the Content

The snapshot generates a `configFile.xml` file at the location specified by the SESCrawlerExport component FeedLoc parameter. XML feeds are created in the subdirectory with the source name; for example, wikis. Performing a snapshot can take some time depending on the number of items you have stored on the Content Server instance and how many sources you are generating.

> **Note:**
>
> It is important to take a snapshot before the first crawl or any subsequent full crawl of the source.

To take a snapshot of the content:

1. Log on to WebCenter Content as a system administrator.
2. From the Administration drop-down list, select **SESCrawlerExport**.
3. Select **All sources**, and click **Take Snapshot**.



Creating a Discussions Crawl Source

This option is available only for portals upgraded from prior releases that use Discussions. To create a crawl source to crawl discussion forums and announcements:

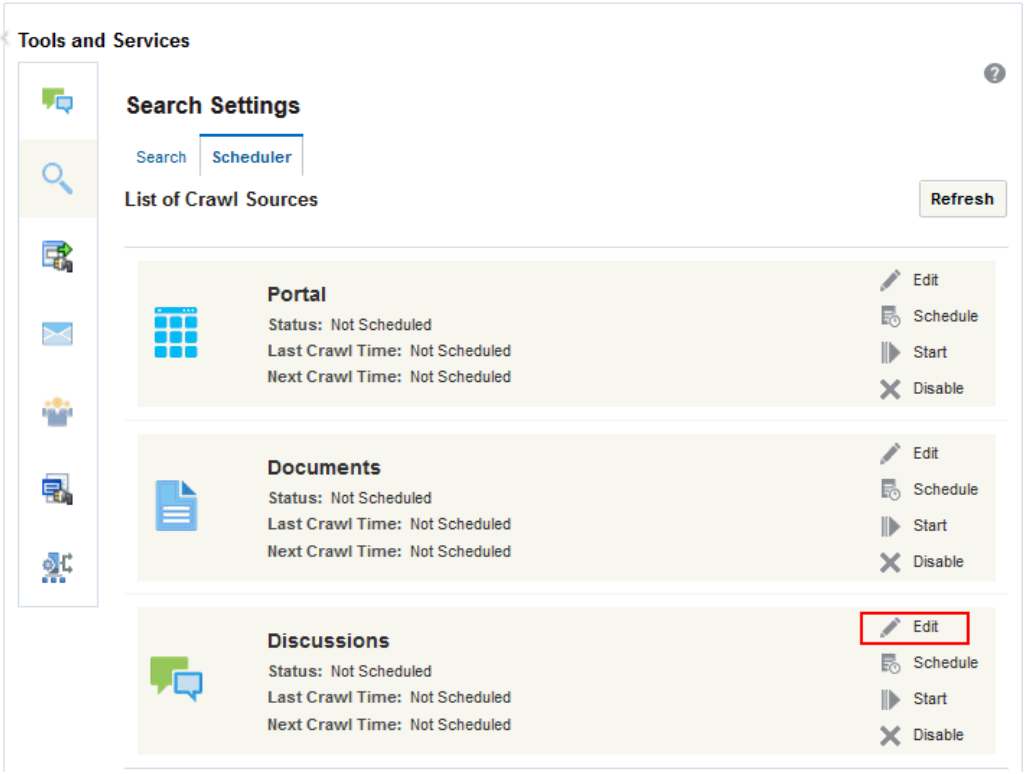
1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Tools and Services**.

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

   `http://host:port/webcenter/portal/admin/settings/tools`

2. Click the icon for Search to open the Search Settings page.
3. On the **Scheduler** tab, select the **Discussions** crawl source and click **Edit**.

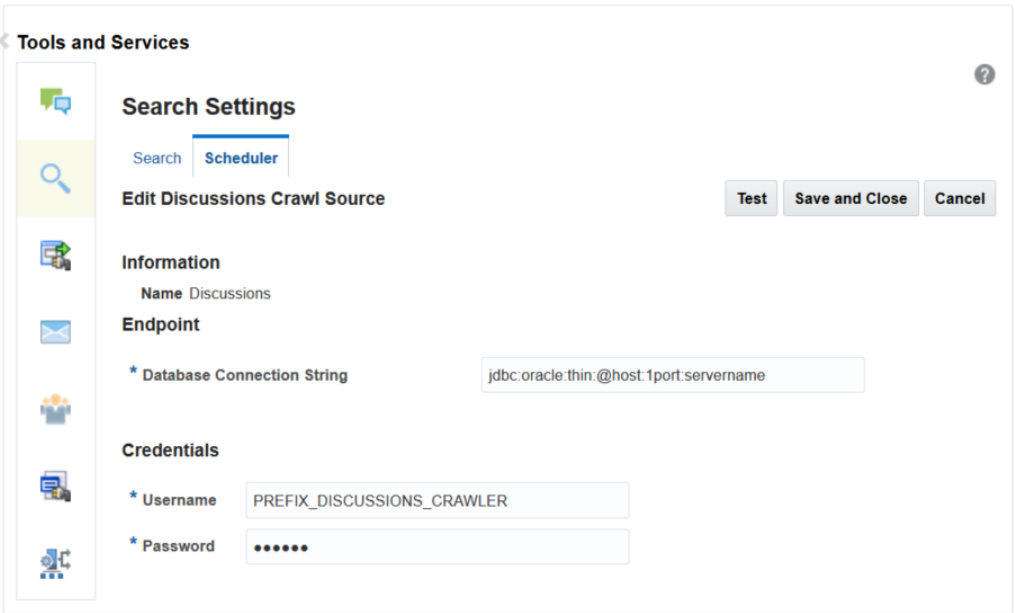
Figure 10-21 Elasticsearch Setting Scheduler



4. On the Edit Discussions Crawl Source page, modify the following source parameter as desired:

   - **Database Connection String**: `jdbc url` of the Discussions schema. The format of the string is `jdbc:oracle:thin@host:port:servername`.

Figure 10-22 Elastic Search Edit Document Crawl Source



5. Enter the Discussions crawler schema user name (for example, `Prefix_Discussions_Crawler`) and password.
6. Click **Test** to test the connection.
7. Click **Save and Close** to save the changes.

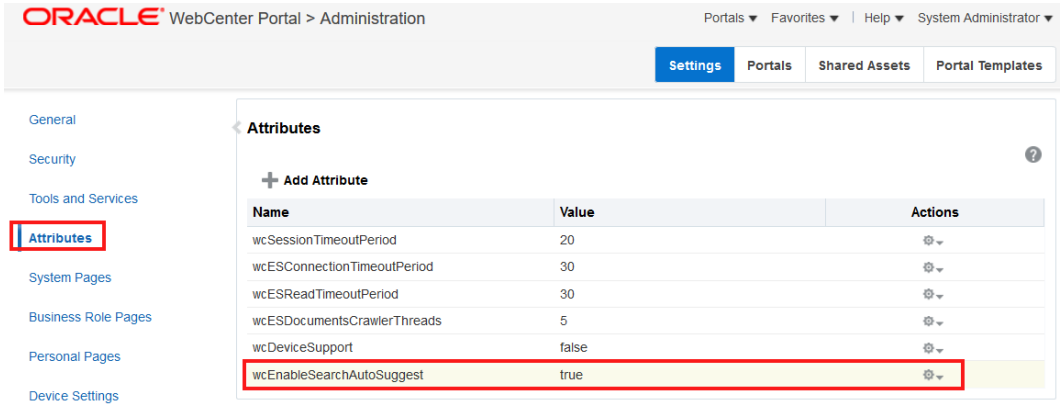
Enabling AutoSuggest in WebCenter Portal

In WebCenter Portal, you can enable autosuggest, so that a list of suggested keywords appear as you type a keyword in the search field. The suggested keywords are based on the portal that you recently accessed.

The attribute, `wcEnableAutoSuggest`, is used to configure autosuggest. By default, the attribute is set to `false`. You can modify the value of the attribute in the WebCenter Portal Administration - Attributes page. After you modify the value, you must restart the WebCenter Portal server for the changes to take effect.

To modify the default value:

1. Log in to WebCenter Portal as an administrator.
2. In the portal browser, click the Administration tile, then click **Attributes** in the left pane.
3. On the Attributes page, click the **Actions** icon for the `wcEnableAutoSuggest` and select **Edit Attribute**.



4. In the Edit Attribute dialog, modify the attribute value to `true`. To disable, set the attribute value to `false`..
5. Click **OK**.
6. Restart WebCenter Portal.

Configuring Search Custom Attributes

When you search using WebCenter Portal, only certain predefined attributes show up in the search results. WebCenter Portal allows you to see additional attributes in your search results. This can be achieved from the Search Setting page in portal administration, where the Custom Attributes section lets you select which custom search attributes should appear in search results and the order in which they appear. This list in the Search Setting page is driven by `search-service-attributes.xml`. It contains list of all attributes that we crawl for

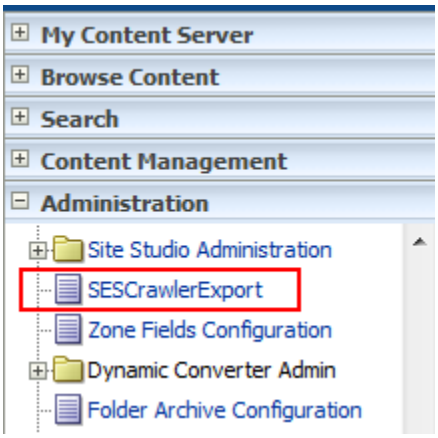
each service. Types in search index is defined by this metadata. You can add a new custom attribute or modify the existing one in the `search-service-attributes.xml` file.

The following procedure describes how to add a new search custom attribute using Document service as an example:

To add a new search custom attribute:

1. Create a custom metadata field. See Adding a Custom Metadata Field in Oracle WebCenter Content .
2. Rebuild the WebCenter Content index. See Rebuilding the WebCenter Content Index.
3. Update the metadata list on Oracle WebCenter Content `SESCrawlerExport` component:
   a. Log on to Oracle WebCenter Content as a system administrator.
   b. On the WebCenter Content home page, expand **Administration** and select **SESCrawlerExport**.

Figure 10-23 WebCenter Content Server Administration Menu



   c. On the SESCrawlerExport Administration page, click **Configure SESCrawlerExport**.

Figure 10-24 SESCrawlerExport Administration Page

d. On the Configure SESCrawlerExport page, append the **Metadata list** with the new custom attribute.

   In this example, Web Site Object Type custom attribute (`xWebsiteObjectType`) is added to the metadata.

```
dDocAuthor,dDocAccount,dDocCreatedDate,dDocCreator,dDocFunction,dDocLastModifie
dDate,dDocLastModifier,dDocName,
dID,dDocType,dFileSize,dFormat,dOriginalName,dOutDate,dRevLabel,dSecurityGroup,
fParentGUID,sceHostname,xClbraRoleList,
xClbraUserList,xComments,xIdcProfile,xLibraryGUID,xWCTags,xWCWorkflowApproverUs
erList,xWCWorkflowAssignment,
xAnnotationDetails,dDocTitle,xRegionDefinition, xWebsiteObjectType
```

4. Add a new custom attribute to the `search-service-attributes.xml` file:

   a. At the WLST command prompt, run the `exportMetadata` WLST command to export the latest `search-service-attributes.xml` file from MDS repository:

```
exportMetadata('webcenter', 'WC_Portal', toLocation='/tmp/es', docs='/oracle/
webcenter/search/scopedMD/
s8bba98ff_4cbb_40b8_beee_296c916a23ed/search-service-attributes.xml')
```

   b. Add the new custom attribute to the `search-service-attributes.xml` file.

   In this example, `Web Site Object Type` is added to the `search-service-attributes.xml` file under the `oracle.webcenter.doclib` service ID:

```
<service id="oracle.webcenter.doclib">
    <attribute name="author" displayNameKey="WC_AUTHOR"
displayName="Author" type="keyword"/>
    <attribute name="doc_author" displayNameKey="WC_DOC_AUTHOR"
displayName="Author" type="keyword" backendAttribute="dDocAuthor"/>
       …
       …
       …
       …
```

```
   <attribute name="wcsecattr" displayNameKey="WC_SECATTR"
displayName="Security Attributes" type="keyword" hidden="true"/>
   <attribute name="wcsecattr_deny"
displayNameKey="WC_SECATTRDENY" displayName="Security Deny
Attributes" type="keyword" hidden="true"/>
   <attribute name="website_object_type"
displayNameKey="WC_xWebsiteObjectType" displayName="Web Site
Object Type"
type="keyword" backendAttribute="xWebsiteObjectType"/>
</service>
```

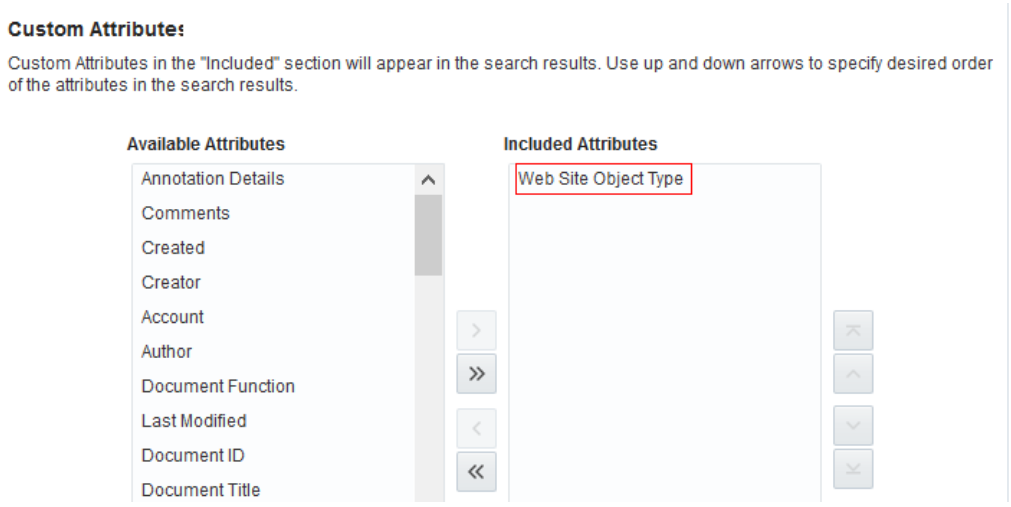
c. Save and import the updated `search-service-attributes.xml` file to the MDS repository using the `importMetada` WLST command:

```
importMetadata('webcenter', 'WC_Portal', fromLocation='/tmp/es', docs='/
oracle/webcenter/search/scopedMD/
s8bba98ff_4cbb_40b8_beee_296c916a23ed/search-service-attributes.xml')
```

5. Access the Search Settings page in portal administration and verify if the new attribute is appearing in the **Available Attributes** under **Custom Attributes** section.

   In this example, Web Site Object is the newly added custom attribute.

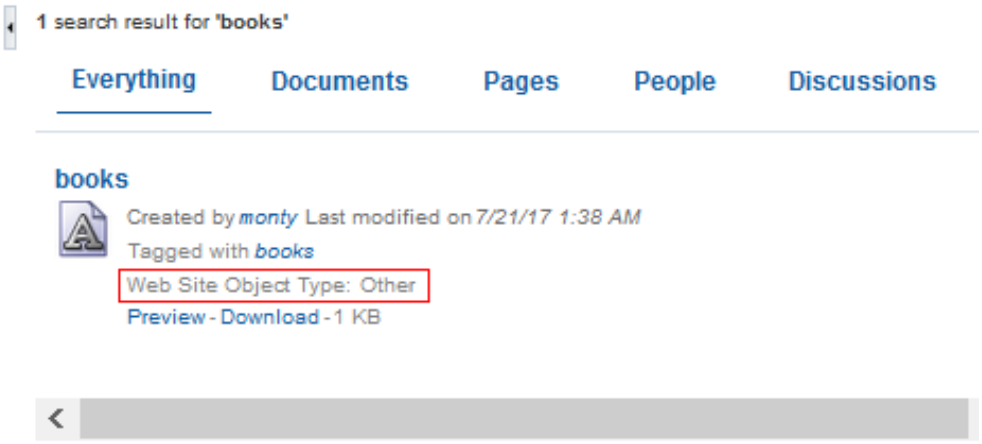
Figure 10-25 Verifying the Newly Added Custom Attribute



6. Select the newly added custom attribute from the list of **Available Attributes** and move it to the **Included Attributes** column so that they appear in search results.
7. Click **Apply**.
8. Navigate to your portal and enter a search term in the global search field and click the search icon.

   The following figure shows search results that include the newly created custom attributes, Web Site Object Type:

Figure 10-26 Search Results with the Created Custom Attribute



Creating Custom Facets

Oracle WebCenter Portal supports faceted search to refine the search results without running a new search. With faceted search, search results are grouped under a predefined category and thus help users to narrow down the search results based on a relevant category. for example, `Author`, `Portal`, `Last Modified date`.

In Oracle WebCenter Portal, by default, certain predefined facets are provided in the Search Setting page. The list of facets is driven by the `search-service-custom-facets.xml` metadata file and each facet in the file is mapped with `search-service-attributes.xml` metadata file.

The following is the sample of the `search-service-custom-facets.xml` metadata file:

```
<custom-facet dataType="keyword" displayName="Author" displayNameKey="ES_FACET_AUTHOR"
mappedSearchAttribute="author" name="Author" itemsToDisplay="20"/>
<custom-facet dataType="keyword" displayName="Service ID"
displayNameKey="ES_FACET_SERVICEID" mappedSearchAttribute="wc_serviceId" name="Service
ID"/>
<custom-facet dataType="keyword" displayName="Portal" displayNameKey="ES_FACET_SCOPE"
mappedSearchAttribute="wc_scopeGuid" name="Scope GUID" itemsToDisplay="20"/>
<custom-facet dataType="keyword" displayName="Tags" displayNameKey="ES_FACET_TAGS"
mappedSearchAttribute="wc_tagWords" name="Tags"itemsToDisplay="20"/>
<custom-facet dataType="keyword" displayName="Mimetype"
displayNameKey="ES_FACET_MIMETYPE" mappedSearchAttribute="mimetype" name="Mimetype"
itemsToDisplay="20"/>
<custom-facet dataType="date" displayName="Last Modified Date"
displayNameKey="ES_FACET_LASTMODIFIED" mappedSearchAttribute="lastmodified"
name="Last Modified Date" />
```

where,

- `name` is the name of the facet.
- `displayNameKey` is the value of the custom facet metadata field.
- `displayName` is the display name of the facet.

> **Note:**
>
> If your business is supported in multiple languages, you can translate the newly added custom facets to the desired languages. See Translating Strings for Search Facets.

- `mappedSearchAttribute` is used to map the custom facet with the search attribute. This value should be same as `name` attribute value in the `search-service-attributes.xml` metadata file.
- `dataType` is the type of data. The available data types are `keyword`, `keyword_text`, and `date`. This value should be same as the type value in the `search-service-attributes.xml` metadata file. The data type `keyword` should be used when the user wants to do a case-sensitive search for the value of the custom facet. The data type `keyword_text` should be used when the user wants to do the case-insensitive search for the value of the custom facet.
- `itemsToDisplay` is an optional attribute that defines the maximum number of items to be displayed under a facet. If the value is not specified, the default value is configured using option **Administration > Tools and Services > Search** in WebCenter Portal.

> **Note:**
>
> This attribute is supported only if dataType `keyword` is used.

Based on your business needs, you can add or modify the list of the facets in the `search-service-custom-facets.xml` metadata file for any of the services available in Oracle WebCenter Portal, for example, you can add custom facets for documents, people and other services listed in the `search-service-custom-facets.xml` metadata file.

This section shows how to add a custom facet for the document service. To add custom facet for the document service, you need to first add a custom metadata field in Oracle WebCenter Content, and then rebuild the content index.

The following steps shows you how to add the custom facet for the document service:

1. Adding a Custom Metadata Field in Oracle WebCenter Content
2. Rebuilding the WebCenter Content Index
3. Configuring the SESCrawlerExport Component
4. Configuring the Search Setting Metadata
5. Configuring the Search Custom Facet Metadata
6. Update the Metadata for the Document in Your Portal
7. Configuring the File Exclusion in Search Results

Adding a Custom Metadata Field in Oracle WebCenter Content

To add a custom facet for the document service, you need to begin with creating a metadata field in Oracle WebCenter Content. In this example, we'll create a new metadata field and name it as *CustomMetadataField*.

To add a custom metadata field in Oracle WebCenter Content:

1. Log in to Oracle WebCenter Content server as a system administrator.
2. On the WebCenter Content home page, expand **Administration** and select **Admin Applets**.
3. On the Administration Applets page, click **Configuration Manager**.
4. On the Configuration Manager page, click the **Information Fields** tab and click **Add** to add a new custom metadata field.
5. In the Add Metadata Field Name dialog, enter the name of the metadata. For example, *CustomMetadataField* and click **OK**.

> **Note:**
>
> The metadata field name should be unique and the maximum field length is 29 characters. Use only letters, numbers, and underscores (_). Do not use spaces or other special characters.

6. In the Add Metadata Field dialog, edit the following information:
   - **Field Type:** Select **Long Text** from the drop-down list.
   - **Enable on User Interface:** Select **Enabled** to display the field. If you don’t select the option, the field is hidden in the search pages.
   - **Enable for Search Index:** Select **Indexed** for the metadata field to be indexed for searching. If you don’t select the option, the field is not indexed nor does it appear on search pages.
7. Click **OK**.
8. In the Configuration Manager page, click **Update Database Design** and click **OK** to confirm.

Rebuilding the WebCenter Content Index

In Oracle WebCenter Content, after you've created the new metadata field, you need to rebuild the collection and update the search index using the **Repository Manager** utility.

To rebuild and update the search index in Oracle WebCenter Content:

1. Log in to Oracle WebCenter Content server as a system administrator.
2. On the Oracle WebCenter Content home page, expand **Administration** and select **Admin Applets.**
3. On the Administration Applets page, click **Repository Manager.**
4. On the Repository Manager page, click the **Indexer** tab.
5. Under Collection Rebuild cycle section, click **Start** and then click **OK** to proceed.

> **Note:**
>
> Rebuilding the search index is a time-consuming process.

6. Next, configure the `SESCrawlerExport` component with the new custom metadata field. See Configuring the SESCrawlerExport Component.

Configuring the SESCrawlerExport Component

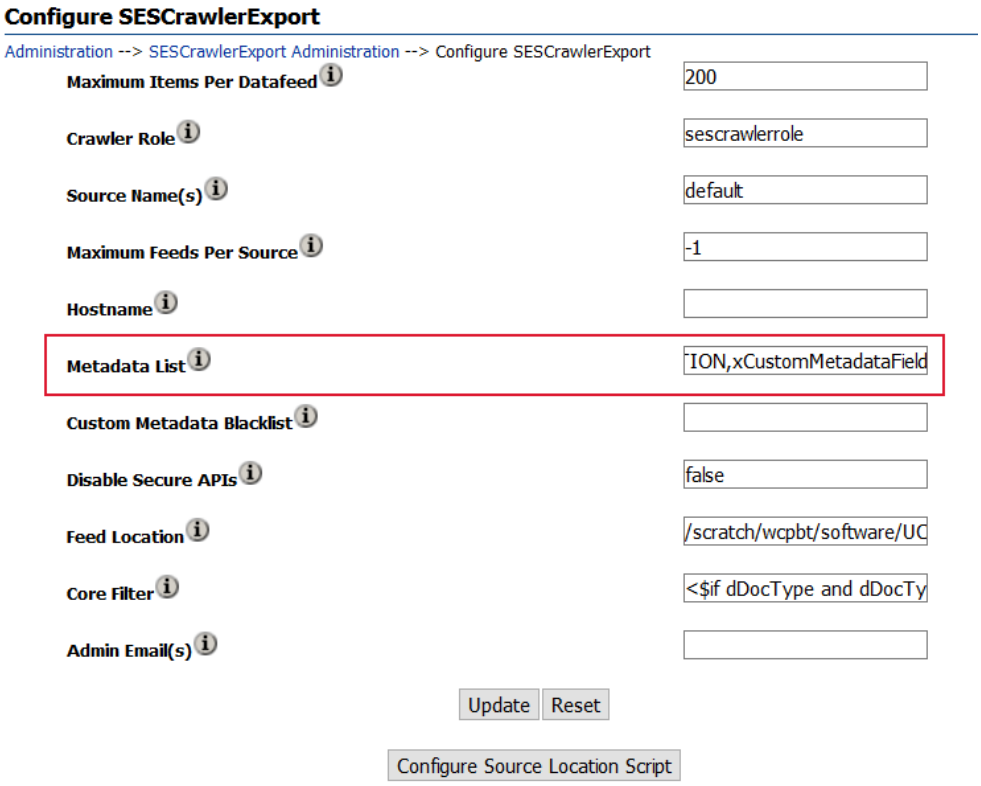
You need to update the Oracle WebCenter Content `SESCrawlerExport` component with the newly created metadata field.

To update the `SESCrawlerExport` component:

1. Log in to WebCenter Content server as a system administrator.
2. On the WebCenter Content home page, expand Administration and select **SESCrawlerExport**.
3. On the SESCrawlerExport Administration page, click **Configure SESCrawlerExport**.
4. On the Configure SESCrawlerExport page, append the **Metadata List** with the newly created metadata field. For example, `xCustomMetadataField`.

   The following figure shows the Configure `SESCrawlerExport` page, where the newly created metadata field is added to the Metadata List.

Figure 10-27 Configure SESCrawlerExport Page



5. Click **Update**.
6. Next, add the new search metadata field to the search-service-attributes.xml file. See Configuring the Search Setting Metadata .

Configuring the Search Setting Metadata

You need to add the defined custom attribute in Oracle WebCenter Portal. For the new custom attribute to appear in the search settings page, you need to add or update the `search-service-attributes.xml`.

To add the defined custom attribute:

1. At the WLST command prompt, run the `exportMetadata` WLST command to export the latest `search-service-attributes.xml` file from MDS repository:

```
exportMetadata(application='webcenter', server='WC_Portal',
toLocation='/tmp/mds_dump', docs='/oracle/webcenter/search/scopedMD/
s8bba98ff_4cbb_40b8_beee_296c916a23ed/search-service-attributes.xml')
```

2. Add the new search attribute field to the `search-service-attributes.xml` file within the section for `id="oracle.webcenter.doclib"`. Set the type field to *keyword.*

```
<service id="oracle.webcenter.doclib">
.......................
.......................
<attribute name="custom_metadata_field"
displayNameKey="WC_CustomMetadataField"
displayName="Custom Metadata Field"  type="keyword"
backendAttribute="xCustomMetadataField"/>
</service>
```

> **Note:**
>
> The `displayName` attribute can be used to provide language support, `displayName` appears in **Custom Attributes** section of Search Settings in Portal.
>
> Oracle WebCenter Portal supports translations of strings to other languages. To translate the `displayName` attribute to other language, see Translating Strings for Custom Search Attributes.

3. Save and import the updated `search-service-attributes.xml` file to the MDS repository using the `importMetada` WLST command:

```
 importMetadata(application='webcenter', server='WC_Portal',
fromLocation='/tmp/mds_dump',docs='/oracle/webcenter/search/scopedMD/
s8bba98ff_4cbb_40b8_beee_296c916a23ed/search-service-attributes.xml')
```

4. Next, create a new facet and map that facet to the search attribute. See Configuring the Search Custom Facet Metadata.

Configuring the Search Custom Facet Metadata

In WebCenter Portal Search Setting page, you can select which facets to display with search results. This list in the Search Setting page is driven by the `search-service-custom-facets.xml` metadata file. It contains a list of facets used in WebCenter Portal. Each facet in the search-`service-custom-facets.xml` metadata file is mapped with a custom attribute using the **mappedSearchAttribute** attribute.

To configure the search custom facet metadata:

1. At the WLST command prompt, run the `exportMetadata` WLST command to export the latest `search-service-custom-facets.xml` metadata file from MDS repository:

   ```
   exportMetadata(application='webcenter', server='WC_Portal',
   toLocation='/tmp/mds_dump',docs='/oracle/webcenter/search/scopedMD/
   s8bba98ff_4cbb_40b8_beee_296c916a23ed/search-service-custom-
   facets.xml')
   ```

2. Open the `search-service-custom-facets.xml` file and create a new facet, and then map the facet to the search attribute.

   ```
   ?xml version='1.0' encoding='UTF-8'?>
   <search-service-custom-facets
   xmlns="http://xmlns.oracle.com/webcenter/search/customfacets">
   ..........................
   <custom-facet name="CustomFacet" dataType="keyword"
   displayNameKey="WC_CustomFacet"
    displayName="Custom Facet"
   mappedSearchAttribute="custom_metadata_field"/>
   </search-service-custom-facets>
   ```

   Where

   - `custom-facet name` is the name of the custom facet that you want to create.

     > **Note:**
     >
     > `dataType` must be `keyword`, and the `displayName` attribute should be used to provide the language support, `displayName` appears in **Facets** section of Search Settings in portal.
     >
     > Oracle WebCenter Portal supports translations of strings to other languages. To translate the `displayName` attribute to other language, see Translating Strings for Search Facets.

   - `mappedSearchAttribute` is the attribute used to map the facet with custom attrbute.

> **Note:**
>
> The `mappedSearchAttribute` attribute in the `search-service-custom-facets.xml` metadata file should contain the same value as specified in the `name` attribute of the `search-service-attributes.xml` metadata file. In this example, use `custom_metadata_field`.

3. Save and import the `search-service-custom-facets.xml` metadata file to the MDS repository using the `importMetada` WLST command:

   ```
   importMetadata(application='webcenter', server='WC_Portal',
   fromLocation='/tmp/mds_dump',docs='/oracle/webcenter/search/scopedMD/
   s8bba98ff_4cbb_40b8_beee_296c916a23ed/search-service-custom-facets.xml'
   ```

4. Log in to WebCenter Portal.
5. Access the Search Settings page in portal administration and verify if the new attribute is appearing in the **Facets** and **Custom attributes** section.
6. Select and move the newly added facet to the **Included Facets** section, then click **Apply**.

   In this example, select `Custom Facet`.
7. Select and move the newly added custom attribute to the **Included Attributes** section, then click **Apply**.

   In this example, select `CustomMetadata Field`.

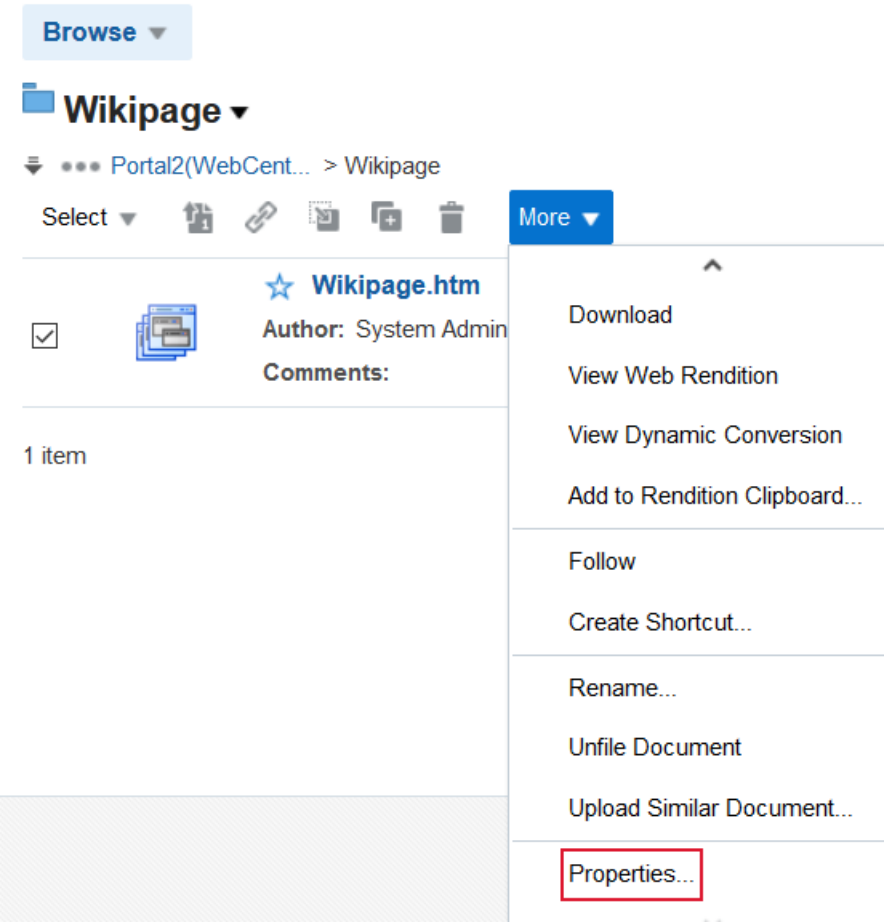
Update the Metadata for the Document in Your Portal

Add the newly created metadata field value to the documents in your portal. In this example, `CustomMetadataField`.

To update metadata field value for document:

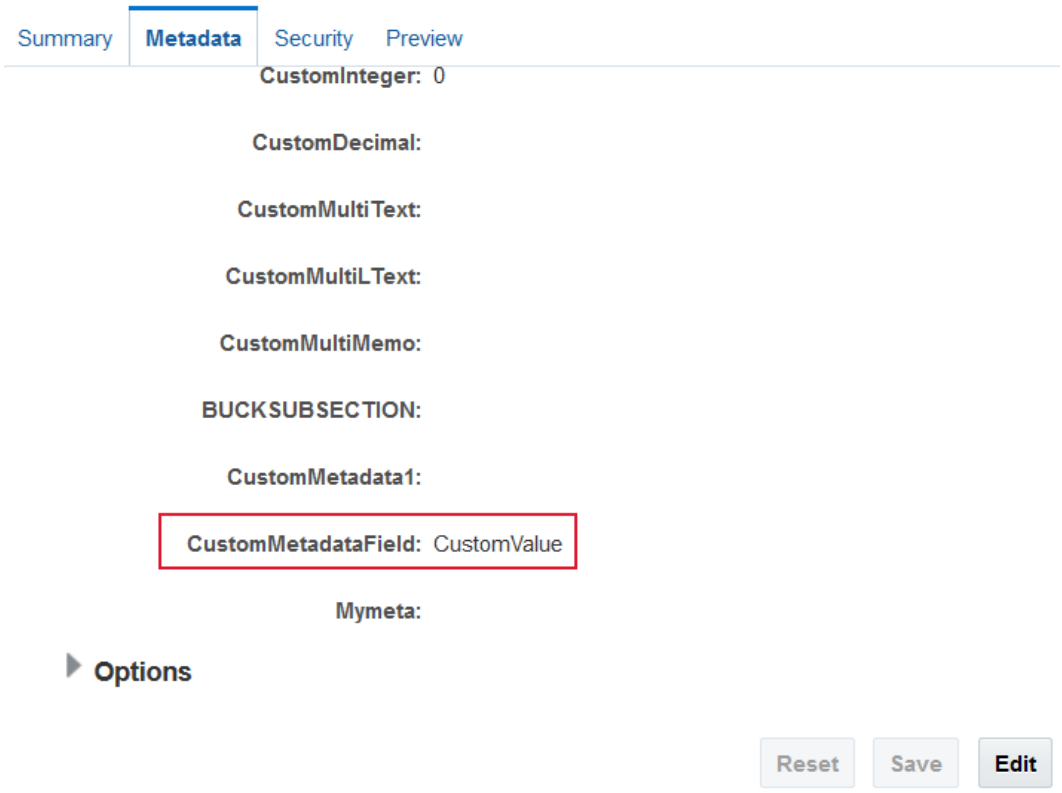
1. In your portal, navigate to the document page.
2. Select the document that you want to edit and click **Properties** from the **More** drop-down menu.

Figure 10-28 Content Manager



3. In the Metadata tab, locate the newly added metadata field, `CustomMetadataField`, and click **Edit**.
4. Add a value for the newly created metadata field and click **Save**.

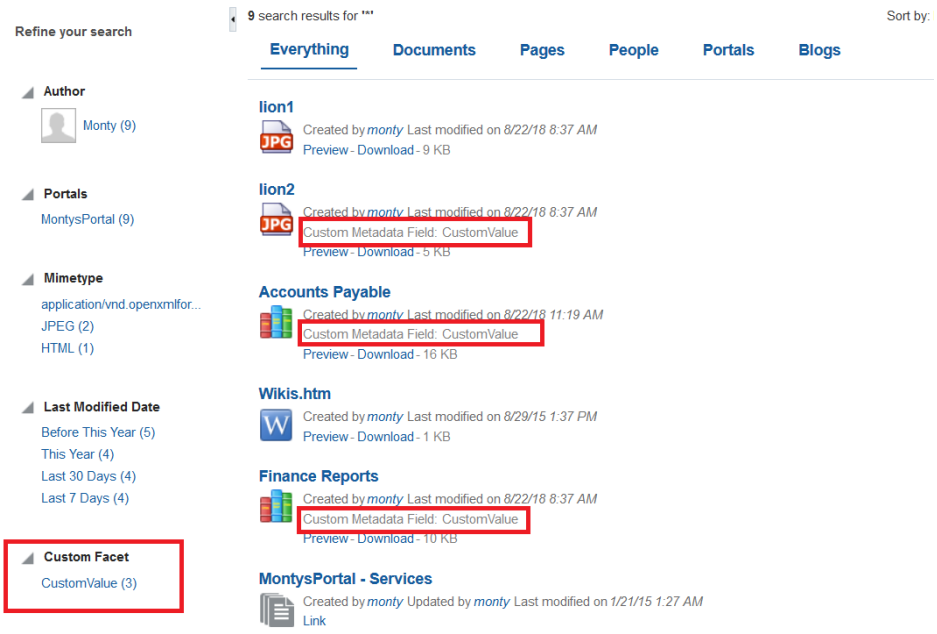
Figure 10-29 Edit Document Metadata Tab



5. Perform an incremental crawl for the new facet to appear in the Search Settings page. See Manually Starting an Incremental Crawl.
6. Navigate to your portal and enter a search term in the global search field and click the search icon.

   The following figure shows search results that include the newly created facet, `Custom Facet`:

Figure 10-30 Search Results with the Created Facet



Configuring the File Exclusion in Search Results

You can configure to exclude certain file types from the search results. In order to exclude the file types, the file types have to be specified in the metadata file `search-service-exclude-file-types.xml`.

The following steps should be performed to configure the file exclusion in search results:

1. Export Metadata Service Document
2. Update the File Types to Filter
3. Import Metadata Service Document
4. Verifying File Exclusion in Search Results

Export Metadata Service Document

The location of the file in MDS is `/oracle/webcenter/search/scopedMD/s8bba98ff_4cbb_40b8_beee_296c916a23ed/search-service-exclude-file-types.xml`.

To export the metadata service document `search-service-exclude-file-types.xml`, enter the following command:

```
exportMetadata(application='webcenter', server='WC_Portal',
toLocation='/tmp/search', docs='/oracle/webcenter/search/scopedMD/
s8bba98ff_4cbb_40b8_beee_296c916a23ed/search-service-exclude-file-
types.xml')
```

In the metadata service document `search-service-exclude-file-types.xml`, you can customize the file types to be excluded.

Update the File Types to Filter

The metadata describes how we can specify the files types which have to be excluded during the search.

```
<?xml version='1.0' encoding='UTF-8'?>
<search-service-exclude-file-types xmlns="http://xmlns.oracle.com/webcenter/
search/excludefiletypes">
  <exclude-file-in-search>
    <fileType extension=".xml" customType="site-studio-file"
description="Exclude Site Studio data files that have extension .xml"/>
    <fileType extension=".wav" description="Exclude all the file types with
extension .wav"/>
    <fileType extension=".jpeg" description="Exclude all the file types with
extension .jpeg"/>
  </exclude-file-in-search>
</search-service-exclude-file-types>
```

The following are a few important points about this metadata:

- The extension of the file which we want to filter from the search has to be entered in this metadata file.
- The `customType` attribute is optional. If we want to exclude the Site Studio data files in the search results, specify the `customType` as `site-studio-file` for extension ".xml".
- The `description` attribute is optional. It explains the details of the files being excluded.

Import Metadata Service Document

To import the metadata service document `search-service-exclude-file-types.xml`, enter the following command:

```
importMetadata(application='webcenter', server='WC_Portal',
fromLocation='/tmp/search', docs='/oracle/webcenter/search/scopedMD/
s8bba98ff_4cbb_40b8_beee_296c916a23ed/search-service-exclude-file-types.xml')
```

Verifying File Exclusion in Search Results

To verify the file types exclusion, click the **Search** icon and specify the file type, for example, *.wav or *.jpeg, and notice that these file types are excluded in the search result. Similarly, search on the .xml file, and notice that the .xml files created in the Site Studio are excluded.

Customizing Search Query Pattern

Oracle WebCenter Portal allows customizing the Search Query Pattern. This can be done by changing the pattern in the Query String Query used for search.

The following steps should be performed to customize the Search Query Pattern:

1. Export Metadata Service Document
2. Modify Query String Query Pattern
3. Import Metadata Service Document

Export Metadata Service Document

To export the metadata service document `search-service-custom-config.xml`, enter the following command:

```
exportMetadata(application='webcenter', server='WC_Portal',
toLocation='/tmp/search', docs='/oracle/webcenter/search/scopedMD/
s8bba98ff_4cbb_40b8_beee_296c916a23ed/search-service-custom-
config.xml')
```

In the metadata service document `search-service-custom-config.xml`, you can customize the query pattern and enable or disable of appending wildcard to the search term.

Modify Query String Query Pattern

The Query String Query pattern is included as a value of the property called `query_string`. The snippet shown below is the default value for the `query_string` property. The `%s` indicates the token that will be replaced by the search term input through the search field in WebCenter Portal. The `%s` can appear any number of times in the `query_string` value and each occurrence of this token will be replaced with the search term. For more details, see https://www.elastic.co/guide/en/elasticsearch/reference/6.8/query-dsl-query-string-query.html.

```
<!-- The query_string query to be executed in Elasticsearch. The value
should be in JSON format. -->
<property name="query_string"><![CDATA[
   "query_string": {
      "query": "(%s)"
   }]]>
</property>
```

Handling Wildcards

Whether to append the wildcard (`*`) to the search term is determined by the property `append_wildcard`. If true, `*` will be appended to the search term.

> **Note:**
>
> Setting `append_wildcard` to true will cause all the results to have the same relevancy score. To have proper relevancy score, you should use the following in the `query_string` pattern: `rewrite: scoring_boolean`

```
<property name="append_wildcard">false</property>
Example:
<property name="query_string"><![CDATA[
   "query_string": {
      "query": "(%s)",
      "rewrite": "scoring_boolean"
```

```
   }]]>
</property>
```

Sample Query String Query Patterns

The following are some of the examples for the Query String Query patterns:

1. Search across all fields and boost the matches in the `title` field by 2.

```
<property name="query_string"><![CDATA[
   "query_string": {
      "query": "(title:%s^2 OR %s)"
   }]]>
</property>
```

2. Search only in `title` and `content` fields.

```
<property name="query_string"><![CDATA[
   "query_string": {
      "fields": ["title", "content"],
      "query": "(%s)"
   }]]>
</property>
```

3. Search for the exact phrase across all fields.

```
<property name="query_string"><![CDATA[
   "query_string": {
      "query": "(\"%s\")"
   }]]>
</property>
```

Examples of Query String Query

The examples given below are with respect to the sample Query String Query patterns mentioned above:

1. If the search term is `school` and the `append_wildcard` value is false, then `%s` will be replaced with `school` and the `query_string` value will be:

```
"query_string": {
      "query": "(title:school^2 OR school)"
   }

"query_string": {
      "fields": ["title", "content"],
      "query": "(school)"
   }

"query_string": {
      "query": "(\"school\")"
   }
```

> **Note:**
>
> If the `append_wildcard` property value is `true`, `%s` will be replaced with `school*` in the above examples.

2. If the search term is `school children` and the `append_wildcard` value is `false`, then `%s` will be replaced with `school children` and the `query_string` value will be:

```
"query_string": {
      "query": "(title:school children^2 OR school children)"
   }

"query_string": {
      "fields": ["title", "content"],
      "query": "(school children)"
   }

"query_string": {
      "query": "(\"school children\")"
   }
```

> **Note:**
>
> If the `append_wildcard` property value is `true`, `%s` will be replaced with `school children*` in the above examples.

Import Metadata Service Document

To import the metadata service document `search-service-custom-config.xml`, enter the following command:

```
importMetadata(application='webcenter', server='WC_Portal',
fromLocation='/tmp/search', docs='/oracle/webcenter/search/scopedMD/
s8bba98ff_4cbb_40b8_beee_296c916a23ed/search-service-custom-
config.xml')
```

Scheduling a Crawl

You can schedule an incremental search crawl or manually start a full crawl or manually start a fast crawl or manually start a reindex crawl. The topics in this section describe how to schedule a crawl and how to start, enable, or disable a crawl.

- Scheduling an Incremental Crawl
- Enabling and Disabling a Scheduled Crawl
- Manually Starting a Full Crawl
- Manually Starting a Fast Crawl
- Manually Starting a Reindex Crawl
- Manually Starting an Incremental Crawl

Scheduling an Incremental Crawl

By default, the crawler is set to manual, but you can specify a different frequency, such as minutes, hourly or daily:

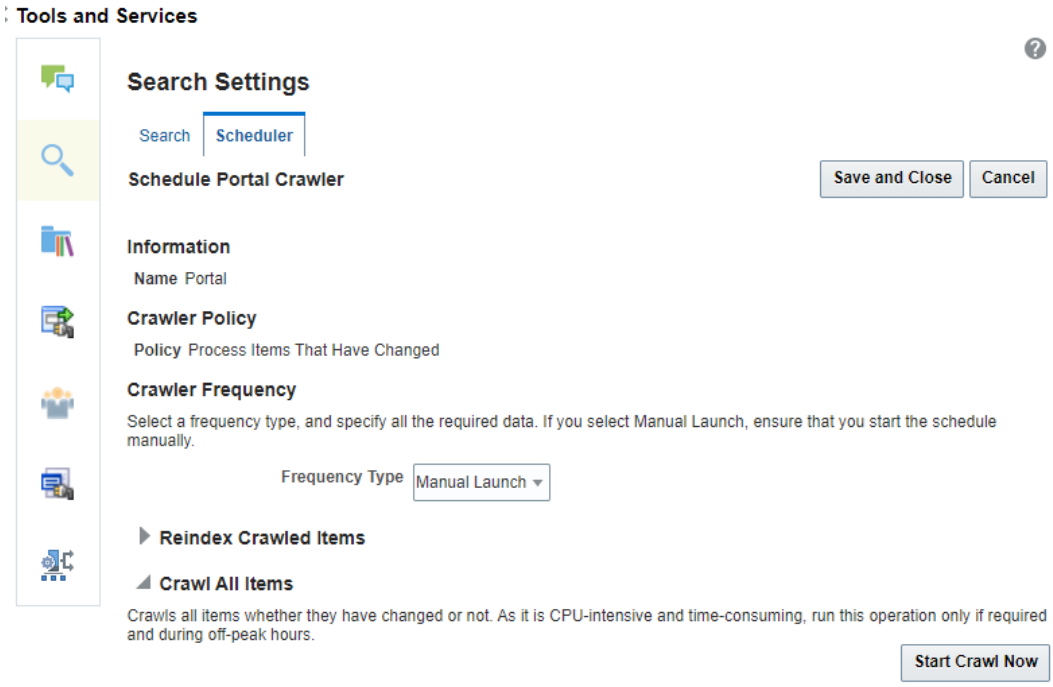
1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Tools and Services**.

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

   `http://host:port/webcenter/portal/admin/settings/tools`

2. Click the icon for Search to open the Search Settings page.
3. On the **Scheduler** tab, select the crawl source and click **Schedule** to open the crawl source Schedule page.

Figure 10-31 Elastic Search Schedule Portal Crawler



4. From the **Frequency Type** list, select the required frequency type.

Frequency Type	Required Data to be Specified
Manual Launch	This is the default crawler frequency type. If you select Manual Launch, ensure that you start the schedule manually.
Minutes	If you select **Minutes**, specify the **Time Between Launches** in minutes. Allowed values: 10 to 59 minutes.

Frequency Type	Required Data to be Specified
Hourly	If you select **Hourly**, specify the **Time Between Launches** in hours. Allowed values: 1 to 23 hours.
Daily	If you select **daily**, specify the **Time Between Launches** in days and specify the schedule launching time. Allowed values: 1 to 99 days and the time from 1 to 12 AM or PM
Weekly	If you select **weekly**, specify the **Time Between Launches** in weeks and specify the schedule launching time. Allowed values: 1 to 12 weeks and the time from 1 to 12 AM or PM.

5. Click **Save and Close**.

Enabling and Disabling a Scheduled Crawl

When a crawl is in progress, you cannot disable it. To enable and disable the schedule defined for a crawl:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Tools and Services**.

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

   `http://host:port/webcenter/portal/admin/settings/tools`

2. Click the icon for Search to open the Search Settings page.
3. On the **Scheduler** tab, select the crawl source that you want to modify and click **Enable** to activate the schedule defined for the crawl or click **Disable** to stop the crawl schedule.

Manually Starting a Full Crawl

You can manually start a full crawl to crawl all items in WebCenter Portal. Start a full crawl only during non-peak times as it is time-consuming. A full crawl must be manually started and cannot be scheduled to run automatically.

To start a full crawl:

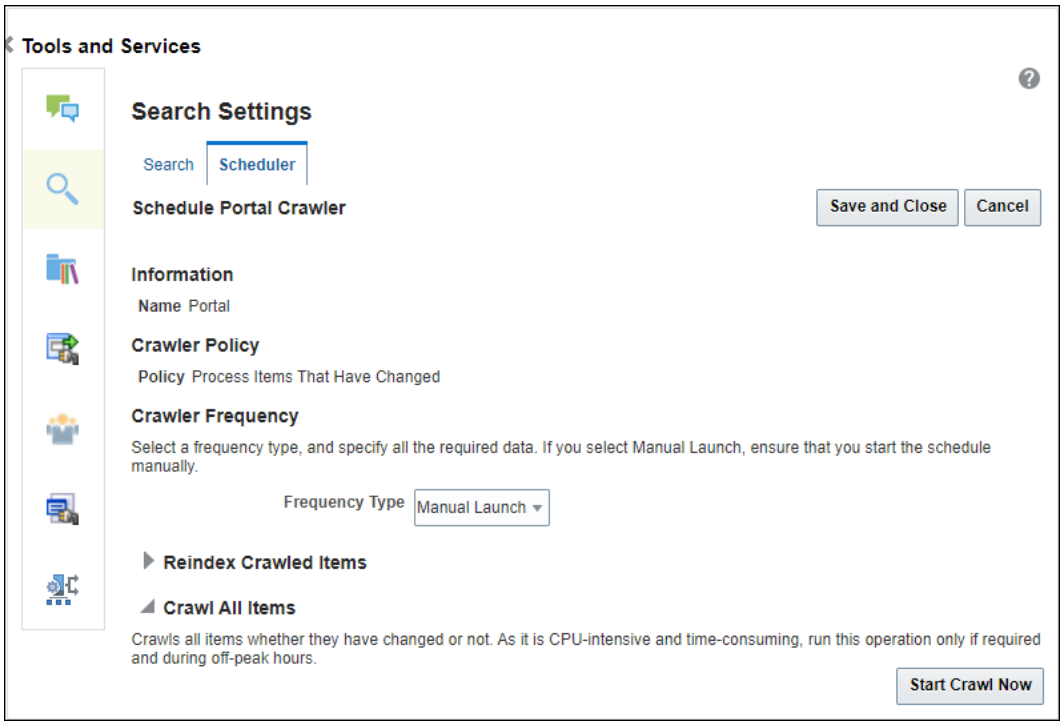
1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Tools and Services**.

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

   `http://host:port/webcenter/portal/admin/settings/tools`

2. Click the icon for Search to open the Search Settings page.
3. On the **Scheduler** tab, select the crawl source and click **Schedule** to open the crawl source Schedule page.

Figure 10-32 Schedule Portal Crawler



4. On the **Scheduler** tab, expand **Crawl All Items** and then click **Start Crawl Now**.

   WebCenter Portal immediately starts a full crawl for the selected crawl source.

Manually Starting a Fast Crawl

You can manually start a fast crawl to crawl only the items that have changed. It crawls only the metadata fields of all the items that have changed. This operation does not crawl the content of the changed documents. This crawl is applicable only to the Documents crawl source. A fast crawl must be manually started and cannot be scheduled to run automatically.

To start a fast crawl:

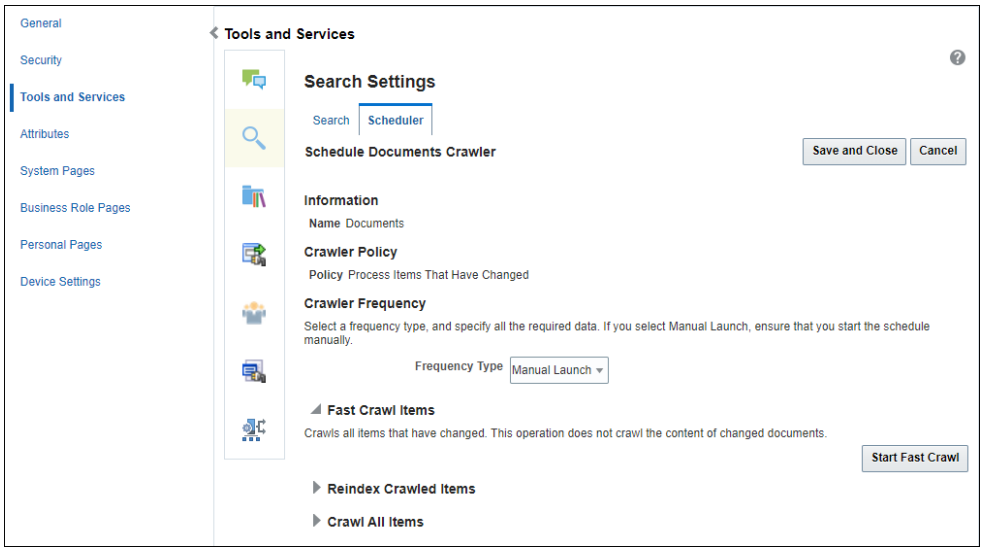
1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Tools and Services**.

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

   `http://host:port/webcenter/portal/admin/settings/tools`

2. Click the icon for Search to open the Search Settings page.
3. On the **Scheduler** tab, select the **Documents** crawl source and click **Schedule** to open the Schedule Documents Crawler page.

Figure 10-33 Schedule Documents Crawler



4. On the **Scheduler** tab, expand **Fast Crawl Items** and then click **Start Fast Crawl**.

   WebCenter Portal immediately starts the fast crawl for the documents.

Manually Starting a Reindex Crawl

You can manually start a reindex crawl when the type of one or more custom search attributes are changed in `search-service-attributes.xml` file. The reindex crawl only copies the crawled items from one index to another. A reindex crawl must be manually started and cannot be scheduled to run automatically. For example, suppose for an attribute `wc_description` the type is defined as `text`. Now, the type is changed to `keyword`. In this scenario, you can start reindex crawl instead of full crawl (as was the case earlier).

To start a reindex crawl:

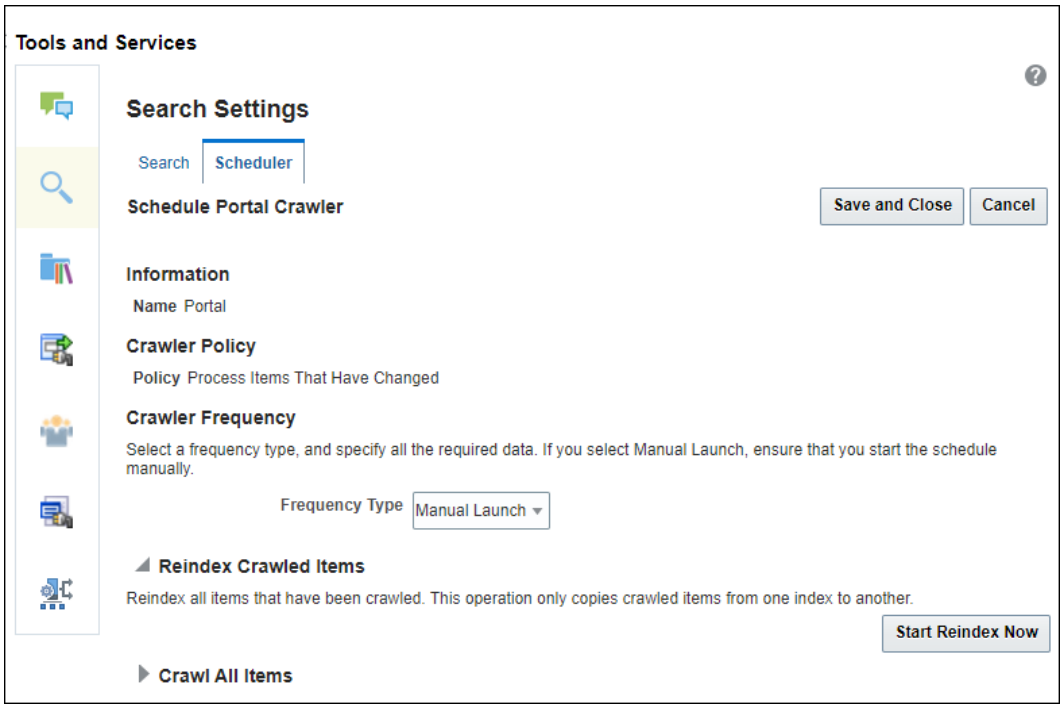
1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Tools and Services**.

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

   `http://host:port/webcenter/portal/admin/settings/tools`

2. Click the icon for Search to open the Search Settings page.
3. On the **Scheduler** tab, select the crawl source and click **Schedule** to open the crawl source Schedule page.

Figure 10-34 Elastic Search Schedule Portal Crawler



4. On the **Scheduler** tab, expand **Reindex Crawled Items** and then click **Start Reindex Now**.

   WebCenter Portal immediately starts a reindex crawl for the selected crawl source.

Manually Starting an Incremental Crawl

You can manually start an incremental crawl to crawl only the items that have been created, updated, or deleted after the last crawl in WebCenter Portal.

To start an incremental crawl:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Tools and Services**.

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

   `http://host:port/webcenter/portal/admin/settings/tools`

2. Click the icon for Search to open the Search Settings page.
3. On the **Scheduler** tab, select the crawl source and click **Start**.

   WebCenter Portal immediately starts an incremental crawl for the selected crawl source.

Customizing Search Settings in WebCenter Portal Administration

You can customize Result Types and Filtering, Search Scope, Facets, and Custom Attributes on the Search Settings page in WebCenter Portal Administration. Portal managers can reset only the search scope for the portals that they manage.

To customize search settings for Elasticsearch:

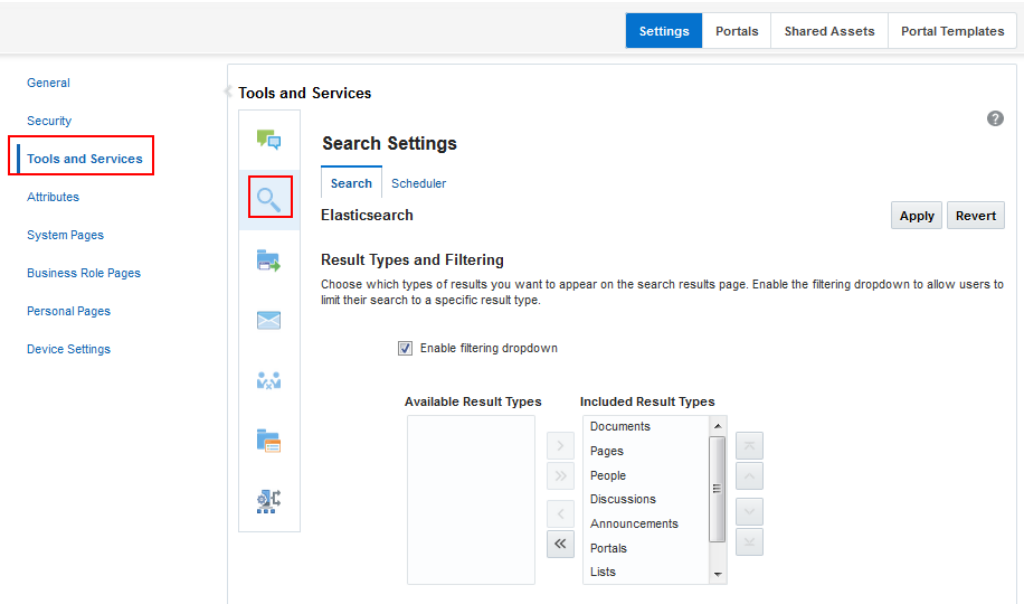
1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Tools and Services**.

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

   `http://host:port/webcenter/portal/admin/settings/tools`

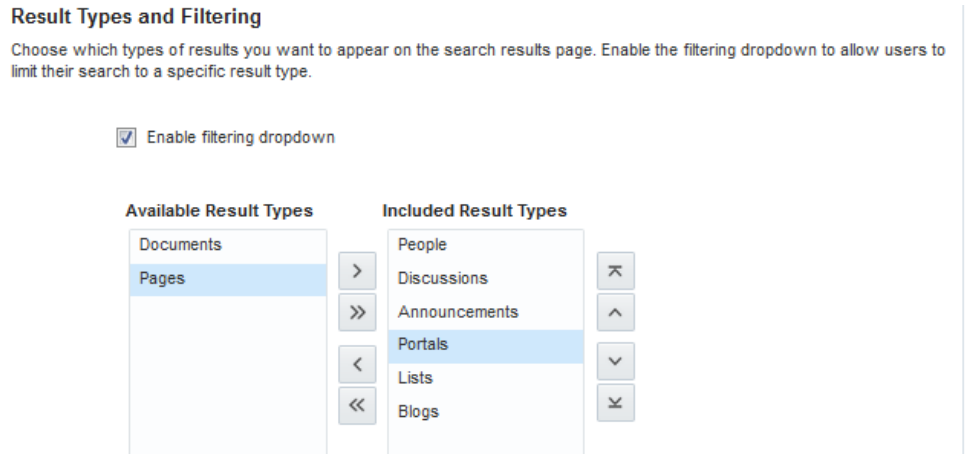
2. Click the icon for Search to open the Search Settings page.

Figure 10-35 Elasticsearch Settings



3. On the **Search** tab, select **Enable filtering dropdown** to enable filtering the search results based on selected services.

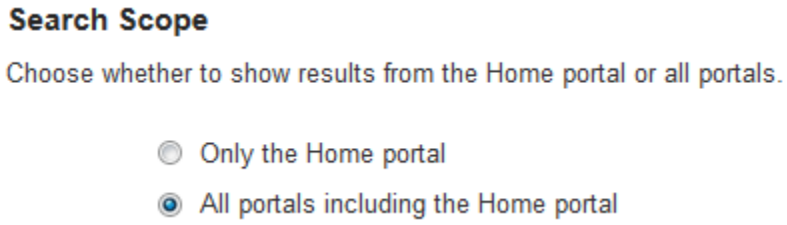
Figure 10-36 Elastic Search Settings-Result Types and Filtering



4. Select the types of results to be included in the search result by moving them between the **Available Result Types** and **Included Result Types**.

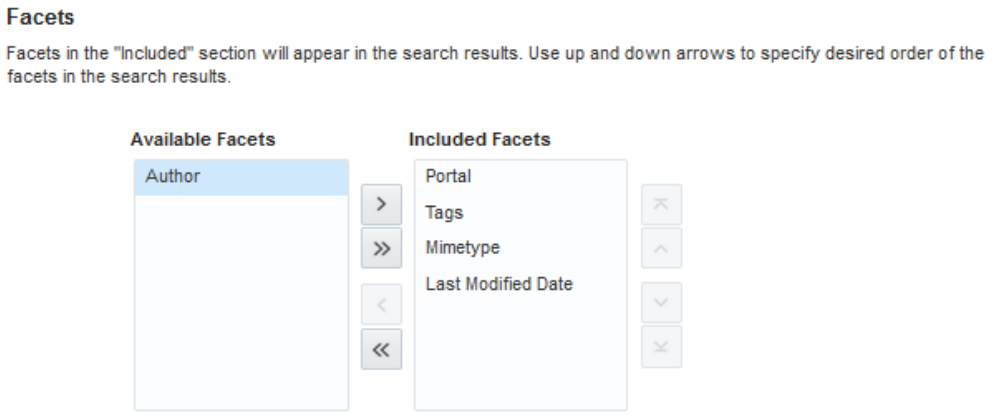
5. Set the search scope to include search results for the Home portal only or all portals (including the Home portal).

Figure 10-37 Elasticsearch Settings - Search Scope



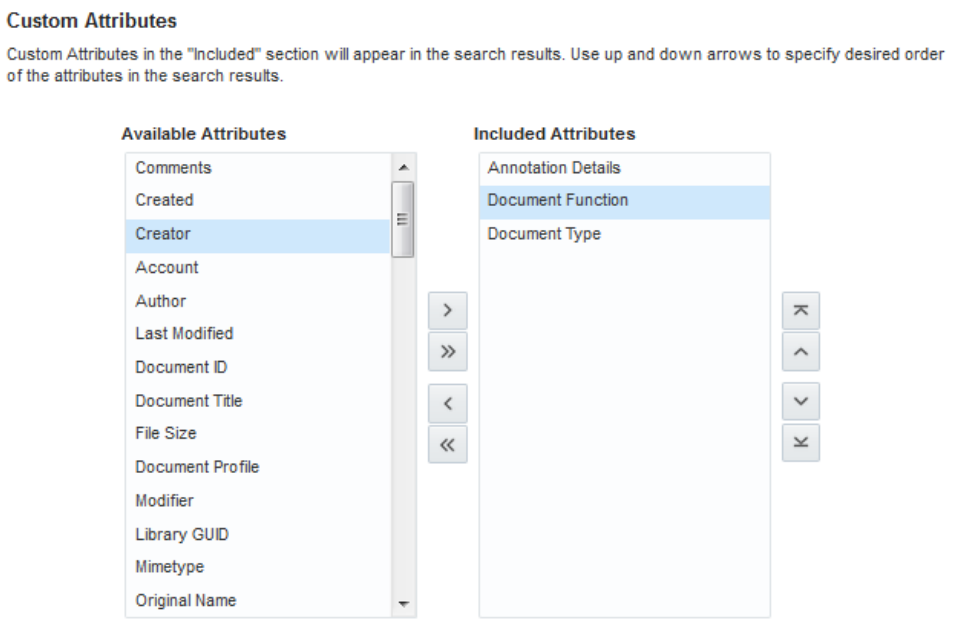
6. Select which facets to display with search results and the order in which they appear by moving them between the **Available Facets** and **Included Facets** lists.

Figure 10-38 Elasticsearch Settings - Facets



7. In the **Custom Attributes** section, select which custom search attributes should appear in search results and the order in which they appear by moving the attributes to the **Included Attributes** section.

Figure 10-39 Elasticsearch Settings - Custom Attributes



8. Click **Apply**.

Modifying the Default Connection Settings for Document Content Crawl Plugin in Elasticsearch Server

After installing Elasticsearch, you can modify the default connection settings for document content crawl plugin using the configuration file.

You can specify the following attributes in the configuration file:

- **es.wcc.connection.timeout** is the connection time-out interval, in seconds. This is the amount of time Elasticsearch server will wait to establish the connection to the WebCenter Content server. The default value is 30 seconds.
- **es.wcc.read.timeout** is the read time-out interval, in seconds. Once Elasticsearch server is connected to the WebCenter Content server, this attribute specifies the amount of time allowed for the WebCenter Content server to respond in a given request. The default value is 30 seconds.
- **es.wcc.max.connection.attempts** is the maximum number of connection attempts to access the WebCenter Content server. The default value is 3.

To modify the default connection settings:

1. In Elasticsearch server, navigate to the `/home/Oracle/product/esHome/esNode/config` directory and create the `webcenter.properties` file
2. In the `webcenter.properties` file, add the following attributes and specify the required value.
   - `es.wcc.connection.timeout`
   - `es.wcc.read.timeout`

- `es.wcc.max.connection.attempts`

For example,

```
es.wcc.connection.timeout=60
es.wcc.read.timeout=60
es.wcc.max.connection.attempts=5
```

3. Save the file and restart Elasticsearch server.

Modifying Search Global Attributes

> **Note:**
>
> The attributes discussed below are of Elastisearch. But these are also applicable if OCI Search Service with OpenSearch is configured.

The attributes `wcESConnectionTimeoutPeriod` and `wcESReadTimeoutPeriod`are used to configure the interaction between WebCenter Portal and Elasticsearch. The `wcESDocumentsCrawlerThreads` attribute is used to configure the number of threads required to process the crawling of documents.

The following are the attributes:

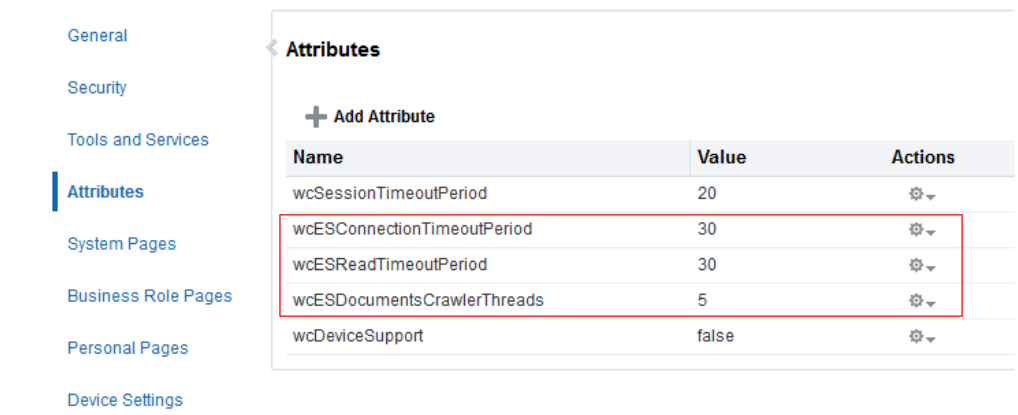
- **wcESConnectionTimeoutPeriod** is the connection timeout interval, in seconds. This is the amount of time WebCenter Portal will wait to establish the connection to the Elasticsearch server. The default value is 30 seconds.
- **wcESReadTimeoutPeriod** is the read timeout interval, in seconds. Once WebCenter Portal is connected to the Elasticsearch server, this specifies the amount of time allowed for the Elasticsearch server to respond in a given request. The default value is 30 seconds.
- **wcESDocumentsCrawlerThreads**: The tasks for crawling the documents for search are handled in threads. This is done by creating a thread pool with a fixed number of threads, where each thread handles the crawl for the documents. The attribute `wcESDocumentsCrawlerThreads` can be used to specify the number of threads used to create a thread pool. The default value is 10. If a thread is not available for a crawl task, the task is in queue, waiting for other task to complete.

You can modify the default value of the attributes in Attributes page in WebCenter Portal administration. After you modify the value, you must restart the WebCenter Portal server for the changes to take effect.

To modify the default value:

1. On the Settings page, click **Attributes**.

Figure 10-40 Attributes for Elasticsearch Settings



2. On the Attributes page, click the **Actions** icon for the attribute and select **Edit Attribute**.
3. In the Edit Attribute dialog, modify the attribute **Value**.
4. Click **OK**.
5. Restart WebCenter Portal.

11

Managing Instant Messaging and Presence

Configure and manage instant messaging and presence (IMP) for WebCenter Portal. Always use Fusion Middleware Control or WLST command-line tool to review and configure back-end tools and services for WebCenter Portal. Any changes that you make to these applications, postdeployment, are stored in MDS metadata store as customizations.

> **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 granted through WebCenter Portal Administration.
>
> For more information about roles and permissions, see Understanding Administrative Operations, Roles, and Tools.

Topics:

- About Instant Messaging and Presence Connections
- Instant Messaging and Presence Server Prerequisites
- Registering Instant Messaging and Presence Servers
- Choosing the Active Connection for Instant Messaging and Presence
- Modifying Instant Messaging and Presence Connection Details
- Deleting Instant Messaging and Presence Connections
- Setting Up Instant Messaging and Presence Defaults
- Testing Instant Messaging and Presence Connections

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 Lync 2010.

> **Note:**
>
> 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. 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 is deployed for changes to take effect.

Instant Messaging and Presence Server Prerequisites

This section describes the Microsoft Lync 2010 prerequisites as the presence server for instant messaging and presence.

This section includes the following subsections:

- Microsoft Lync - Installation
- Microsoft Lync - Configuration
- Microsoft Lync - Security Considerations

Microsoft Lync - Installation

Refer to the Microsoft Lync 2010 documentation for installation information.

Microsoft Lync - Configuration

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 the two topologies:

- Simple Deployment – All components reside on the same box
- Remote Deployment – The proxy application and Microsoft Lync reside on separate boxes

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.

This section includes the following:

- Simple Deployment
- Remote Deployment

Simple Deployment

In a simple topology, install Microsoft Unified Communications Managed API (UCMA) 2.0 on the Lync box. In this topology, WebCenter Portal's Proxy application is deployed in the Internet Information Services (IIS) server hosted on the Lync box. The 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.

This section includes the following:

- Installing UCMA v2.0
- Installing WebCenter Portal's Proxy Application

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/en-us/download/details.aspx?id=9781`
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.

Installing WebCenter Portal's Proxy Application

1. Extract `owc_ocs2007.zip`. The zip file is available in WebCenter Companion Adapters, which you can download from OTN. Navigate to the Downloads page of WebCenter Portal, and download WebCenter Companion Adapters from under the Prerequisites and Recommended Install Process section.

   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
```

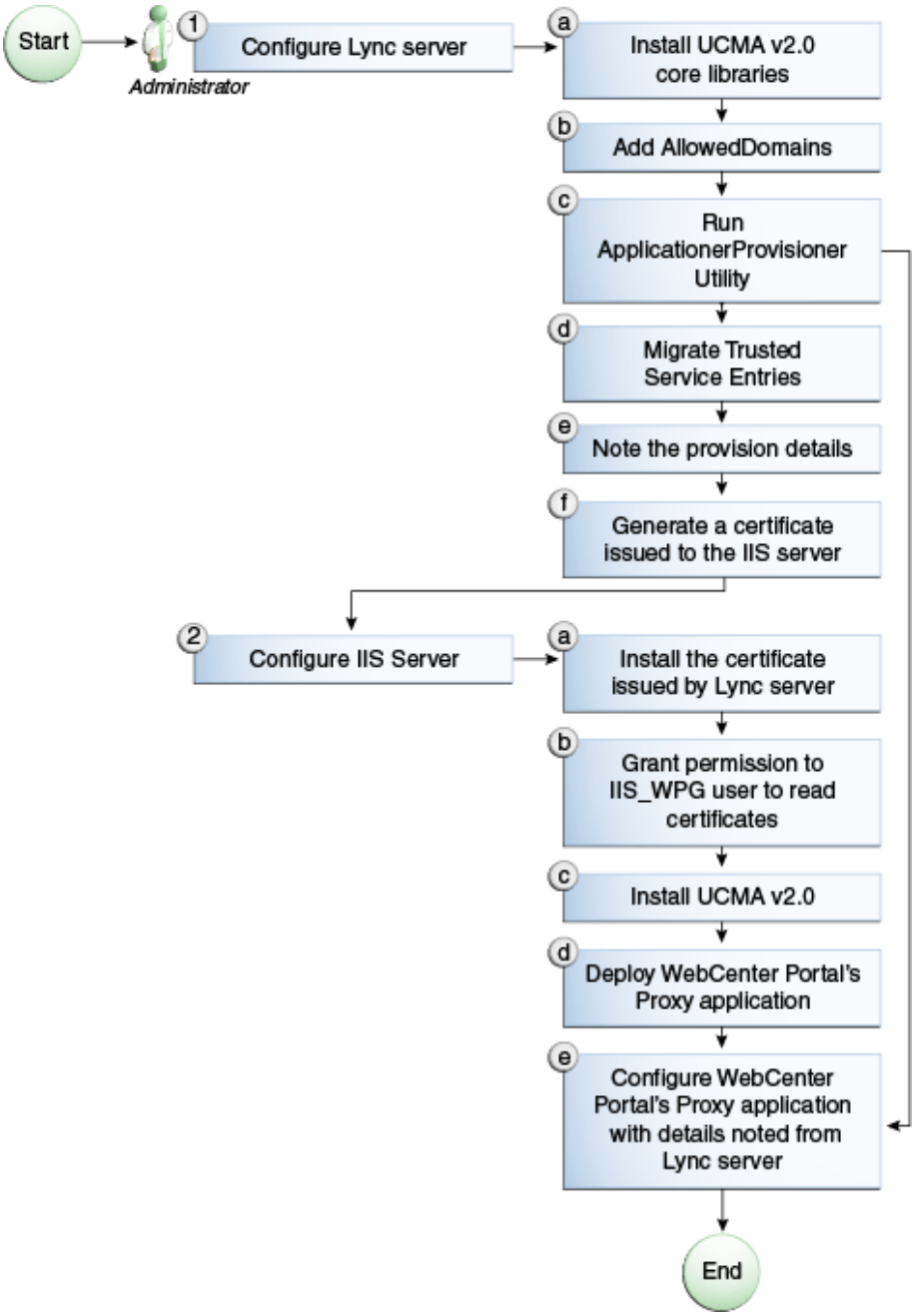
Remote Deployment

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.

Figure 11-1 provides an overview of the steps (including installing UCMA v2.0) to be performed on different deployment entities.

Figure 11-1 Microsoft Lync Configuration - Remote Deployment



The details of these steps are described in the following sections.

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. The Application Provisioner application comes with the UCMA SDK.

a. Download UCMA v2.0 installation from the following location: `http://www.microsoft.com/en-us/download/details.aspx?id=9781`

b. 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`

c. Go to the directory where the setup files were extracted and run `vcredist_x86.exe`.

   Run-time components of Visual C++ Libraries, required for UCMA APIs, are installed.

d. Go to the directory called `Setup` and run `UcmaRedist.msi`.

   UCMA 2.0 assemblies in the GAC are installed.

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.
7. See the next step Provisioning WebCenter Portal's Proxy Application on Lync Server.

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 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 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.

11. See the next step Adding AllowedDomains Using WBemTest.

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**.
15. Go to the next step Migrating Trusted Service Entries Using Topology Builder or PowerShell Cmdlets.

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
   ```

See IIS Server Configuration.

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 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.
   a. Download UCMA v2.0 installation from the following location: `http://www.microsoft.com/en-us/download/details.aspx?id=9781`
   b. 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`
   c. Go to the directory where the setup files were extracted and run `vcredist_x86.exe`.

      Run-time components of Visual C++ Libraries, required for UCMA APIs, are installed.
   d. Go to the directory called `Setup` and run `UcmaRedist.msi`.

      UCMA 2.0 assemblies in the GAC are installed.
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 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 12.2.2.2.5, "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:WebCenterProxyApplication:7m
hSo94PlUK-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.

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 Registering Instant Messaging and Presence Servers Using Fusion Middleware Control.

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:

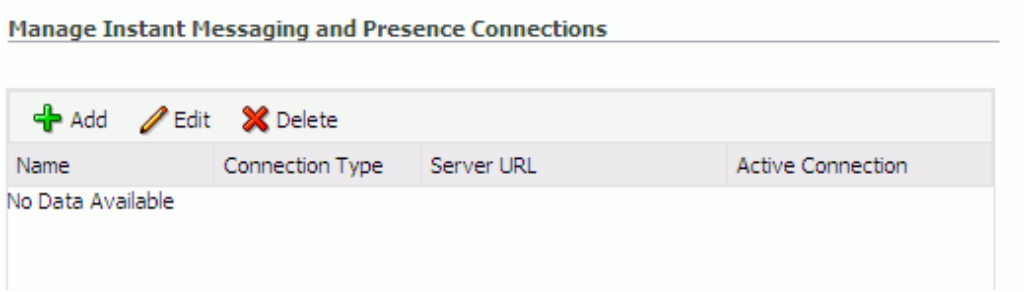
- Registering Instant Messaging and Presence Servers Using Fusion Middleware Control
- Registering Instant Messaging and Presence Servers Using WLST

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.
2. From the **WebCenter Portal** menu, select **Settings** > **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 11-2 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 11-1 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: **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 11-2 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://mylynchost.com:8888`
User Domain	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 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.

Table 11-2 (Cont.) Instant Messaging and Presence Connection - Connection Details

Field	Description
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).

7. Enter a timeout in the Advanced Configuration field.

Table 11-3 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 11-4 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 is deployed.

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 createIMPConnection in *WebCenter WLST Command Reference Reference*.

To configure instant messaging and presence to actively use a new IMP connection, set `default=true`. For more information, see 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 is deployed.

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:

- Choosing the Active Connection for Instant Messaging and Presence Using Fusion Middleware Control
- Choosing the Active Connection for Instant Messaging and Presence Using WLST

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.
2. From the **WebCenter Portal** menu, select **Settings** > **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, restart the managed server on which WebCenter Portal is deployed.

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 setIMPConnection in *WebCenter WLST Command Reference 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 removeIMPServiceProperty in *WebCenter WLST Command Reference Reference*.

> **Note:**
>
> To start using this active connection you must restart the managed server on which WebCenter Portal is deployed.

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:

- Modifying Instant Messaging and Presence Connections Details Using Fusion Middleware Control
- Modifying Instant Messaging and Presence Connections Details Using WLST

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.
2. From the **WebCenter Portal** menu, select **Settings** > **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 11-2.
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.

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 setIMPConnection in *WebCenter WLST Command Reference Reference*.

If additional parameters are required to connect to your presence server, then use the `setIMPConnectionProperty` command. For more information, see setIMPConnectionProperty in *WebCenter WLST Command Reference Reference*.

> **Note:**
>
> To start using the updated (active) connection you must restart the managed server on which WebCenter Portal is deployed.

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 Deleting External Application Connections.

This section includes the following subsections:

- Deleting Instant Messaging and Presence Connections Using Fusion Middleware Control
- Deleting Instant Messaging and Presence Connections Using WLST

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.
2. From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.

3. On the WebCenter Portal Service Configuration page, select **Instant Messaging and Presence**.
4. Select the connection name, and click **Delete**.
5. Restart the managed server on which WebCenter Portal is deployed.

> **Note:**
>
> Before restarting the managed server, mark another connection as active; otherwise, Instant Messaging and Presence is disabled.

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 `deleteConnection` in *WebCenter WLST Command Reference Reference*.

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

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 setIMPServiceProperty in *WebCenter WLST Command Reference Reference*.

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

12

Managing Mail

Configure and manage mail for WebCenter Portal or the "Send Mail" feature to send mail directly from within a portal.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

Always use Fusion Middleware Control or WLST command-line tool to review and configure back-end servers for WebCenter Portal. Any changes that you make to post-deployment, are stored in MDS metatdata store as customizations.

> **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 granted through WebCenter Portal Administration.
>
> For more information about roles and permissions, see Understanding Administrative Operations, Roles, and Tools.
>
> For troubleshooting issues with mail, see Troubleshooting Issues with Mail.

Topics:

- About Mail Server Connections
- Configuration Roadmap for Mail
- Mail Server Prerequisites
- Registering Mail Servers
- Choosing the Active (or Default) Mail Server Connection
- Modifying Mail Server Connection Details
- Deleting Mail Server Connections
- Setting Up Mail Defaults
- Testing Mail Server Connections
- Configuring Send Mail Notifications for WebCenter Portal

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, 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.

Configuration Roadmap for Mail

Use the roadmap in this section as an administrator's guide through the configuration process:

Figure 12-1 and Table 12-1 provide an overview of the prerequisites and tasks required for mail to work in WebCenter Portal.

Figure 12-1 Configuring Mail

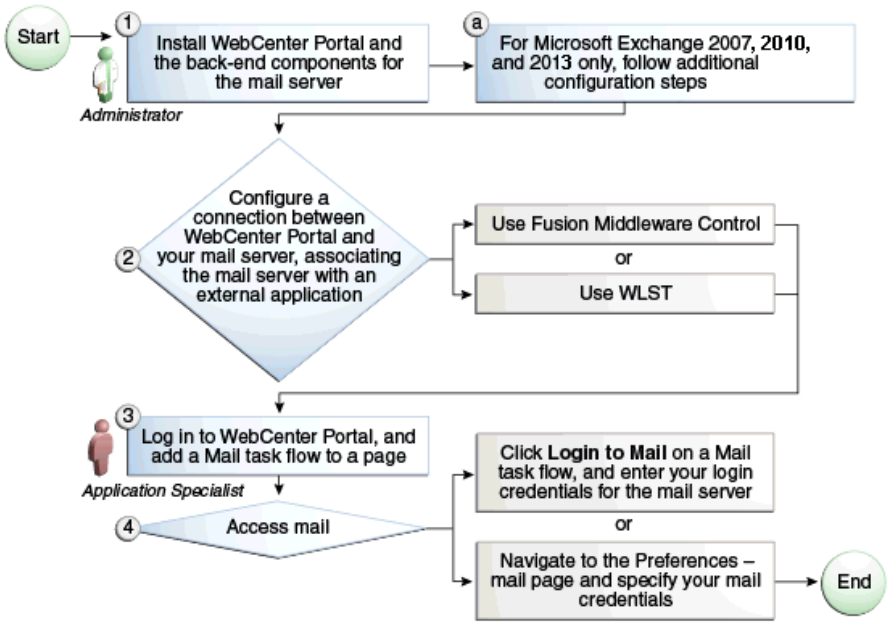


Table 12-1 Configuring Mail for WebCenter Portal

Actor	Task	Link
Administrator	1. Install WebCenter Portal and the required mail server. For Microsoft Exchange Server 2007, 2010, or 2013, perform additional configuration.	See Mail Server - Installation andConfiguring Microsoft Exchange Server 2007, 2010, or 2013 for WebCenter Portal
Administrator	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 • WLST	Registering Mail Servers
Application Specialist	3. Add the Mail task flow to a portal page.	Adding the Mail Task Flow to a Page in *Building Portals with Oracle WebCenter Portal*
Application Specialist/End User	4. Access mail with one of the following methods: • Click Login to Mail on a Mail task flow, and enter your login credentials for the mail server • Navigate to the Preferences - Mail page and specify your mail credentials	See: • Logging in to a Mail Task Flow in *Building Portals with Oracle WebCenter Portal* • Selecting Your Preferred Mail Connection in *Using Portals in Oracle WebCenter Portal*

Mail Server Prerequisites

This section includes the following subsections:

- Mail Server - Installation
- Mail Server - Configuration
- Mail Server - Security Considerations
- Mail Server - Limitations

Mail Server - Installation

See your mail server documentation for installation information.

Mail Server - Configuration

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 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 Adding Users to the Embedded LDAP Identity Store.

Microsoft Exchange 2007, Microsoft Exchange 2010, and Microsoft Exchange 2013 are the only mail servers for which there are configuration prerequisites. If you are working with a different mail server, then you can bypass the rest of this section.

Configuring Microsoft Exchange Server 2007, 2010, or 2013 for WebCenter Portal

The Microsoft Exchange Server 2007, 2010, or 2013 certificate must be added to the WebCenter Portal keystore. This requires the following steps.

1. Obtain the Certificate from the Microsoft Exchange Server
2. Add the Certificate to the WebCenter Portal Keystore
3. Restart the server after the certificate is imported.

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 Server 2007, 2010, or 2013:

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 der2pem in *Command Reference for Oracle WebLogic Server*.

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 Configuring and Exporting the Certificates, for information about adding the certificate to the keystore.

2. Restart the server.

Microsoft Exchange Server Considerations

- The IMAP port is 993 and secured true. SMTP port is 587 and secured true.
- 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:

1. Shutdown the managed server on which WebCenter Portal is deployed.
2. Edit the domain startup script `setDomainEnv` located at:

   UNIX: *`DOMAIN_HOME`*`/bin/setDomainEnv.sh`

   Windows: *`DOMAIN_HOME`*`\bin\setDomainEnv.cmd`
3. 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
   ```
4. Restart the managed server.

Mail Server - Security Considerations

For more information, see 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 12-4.

Mail Server - Limitations

In WebCenter Portal, mail requires a Microsoft Exchange mail server connection to enable automatic WebCenter Portal distribution list management.

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:

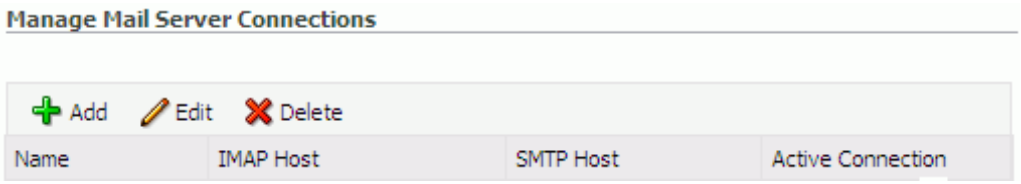
- Registering Mail Servers Using Fusion Middleware Control
- Registering Mail Servers Using WLST

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.
2. From the **WebCenter Portal** menu, select **Settings** > **Service Configuration**.
3. On the WebCenter Portal Service Configuration page, select **Mail Server**.
4. To connect to a new mail server, click **Add**.

Figure 12-2 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 12-2 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.
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.

6. Enter connection details for the mail server.

Table 12-3 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 12-3 (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 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 12-3). See Table 12-7 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 12-4).

   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 12-4 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`.

Table 12-4 (Cont.) LDAP Directory Server Configuration Parameters

Field	Description
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`.
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 12-5 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 12-6 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`. For a list of valid protocol properties, see your mail server documentation. For a list of standard IMAP properties, see the Java Mail APIs: `https://javamail.java.net/nonav/docs/api/com/sun/mail/imap/package-summary.html`

Table 12-6 (Cont.) Additional Mail Connection Properties

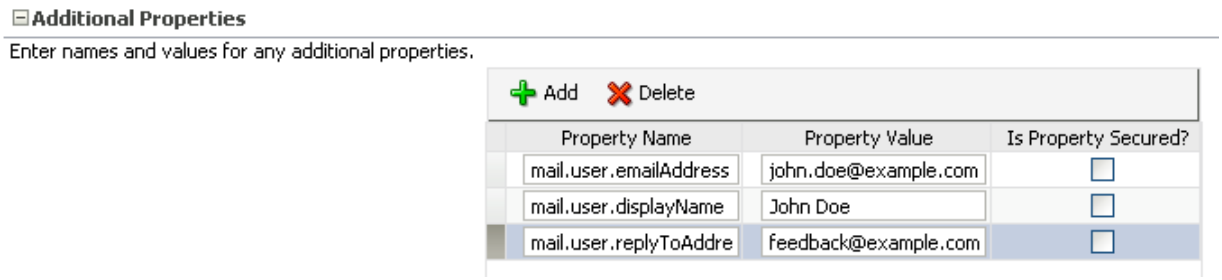
Additional Connection Property	Description
Various SMTP properties	Any valid SMTP connection property. For example, `mail.smtp.timeout`. For a list of valid protocol properties, see your mail server documentation. For a list of standard SMTP properties, see the Java Mail APIs: `https://javamail.java.net/nonav/docs/api/com/sun/mail/smtp/package-summary.html`

If additional parameters are required to connect to the mail server, expand **Additional Properties** and enter details as required.

Table 12-7 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 12-3 Additional Properties for Mail Connection



10. Click **OK** to save this connection.
11. To start using the new (active) connection, restart the managed server on which WebCenter Portal is deployed.

Registering Mail Servers Using WLST

Use the WLST command `createMailConnection` to create a mail server connection. For command syntax and examples, see createMailConnection in *WebCenter WLST Command Reference 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 offers this feature on its default self-registration page.

For command syntax and examples, see setMailConnectionProperty in *WebCenter WLST Command Reference Reference*.

To configure mail to use the new mail server connection as its default connection, set `default=true`. For more information, see Choosing the Active (or Default) Mail Server Connection Using WLST.

> **Note:**
>
> To start using new connections you must restart the managed server on which WebCenter Portal is deployed.

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:

- Choosing the Active (or Default) Mail Server Connection Using Fusion Middleware Control
- Choosing the Active (or Default) Mail Server Connection Using WLST

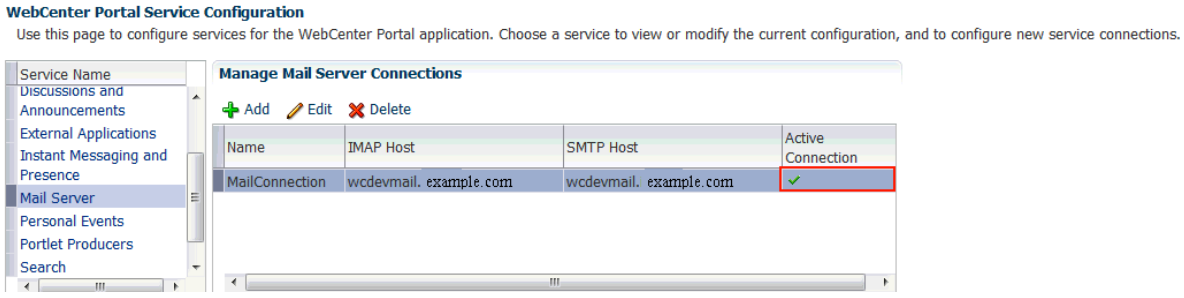
Choosing the Active (or Default) Mail Server Connection Using Fusion Middleware Control

To change the default connection:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal.
2. From the **WebCenter Portal** menu, select **Settings** > **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 12-4 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.

Choosing the Active (or Default) Mail Server Connection Using WLST

Use the WLST command `setMailConnection` with `default=true` to make an existing mail server connection the default connection for mail. For command syntax and examples, see setMailConnection in *WebCenter WLST Command Reference Reference*.

A connection does not cease to be the default connection for mail if you change the default argument from `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 removeMailServiceProperty in *WebCenter WLST Command Reference Reference*.

> **Note:**
>
> To start using the active connection you must restart the managed server on which WebCenter Portal is deployed.

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:

- Modifying Mail Server Connection Details Using Fusion Middleware Control
- Modifying Mail Server Connection Details Using WLST

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 Navigating to the Home Page for WebCenter Portal
2. From the **WebCenter Portal** menu, select **Settings** > **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.

Table 12-8 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.
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 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 12-3). See Table 12-7 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.

6. Click **OK** to save your changes.
7. To start using updated connection details, restart the managed server on which WebCenter Portal is deployed.

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 setMailConnection in *WebCenter WLST Command Reference Reference*.

If additional parameters are required to connect to your mail server, use the `setMailConnectionProperty` command. For more information, see setMailConnectionProperty in *WebCenter WLST Command Reference Reference*.

> **Note:**
>
> To start using the updated connections you must restart the managed server on which WebCenter Portal is deployed.

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 Deleting External Application Connections.

This section includes the following subsections:

- Deleting a Mail Connection Using Fusion Middleware Control
- Deleting a Mail Connection Using WLST

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.
2. From the **WebCenter Portal** menu, select **Settings** > **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, restart the managed server on which WebCenter Portal is deployed.

> **Note:**
>
> Before restarting the managed server, mark another connection as active; otherwise, mail is disabled.

Deleting a Mail Connection Using WLST

Use the WLST command `deleteConnection` to remove a mail server connection. For command syntax and examples, see deleteConnection in *WebCenter WLST Command Reference Reference*.

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 Publishing Portal Mail in a Discussion Forum in *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 setMailServiceProperty in *WebCenter WLST Command Reference Reference*.

Testing Mail Server Connections

Confirm that the mail server is running by connecting to the server using any client, such as Thunderbird or Outlook.

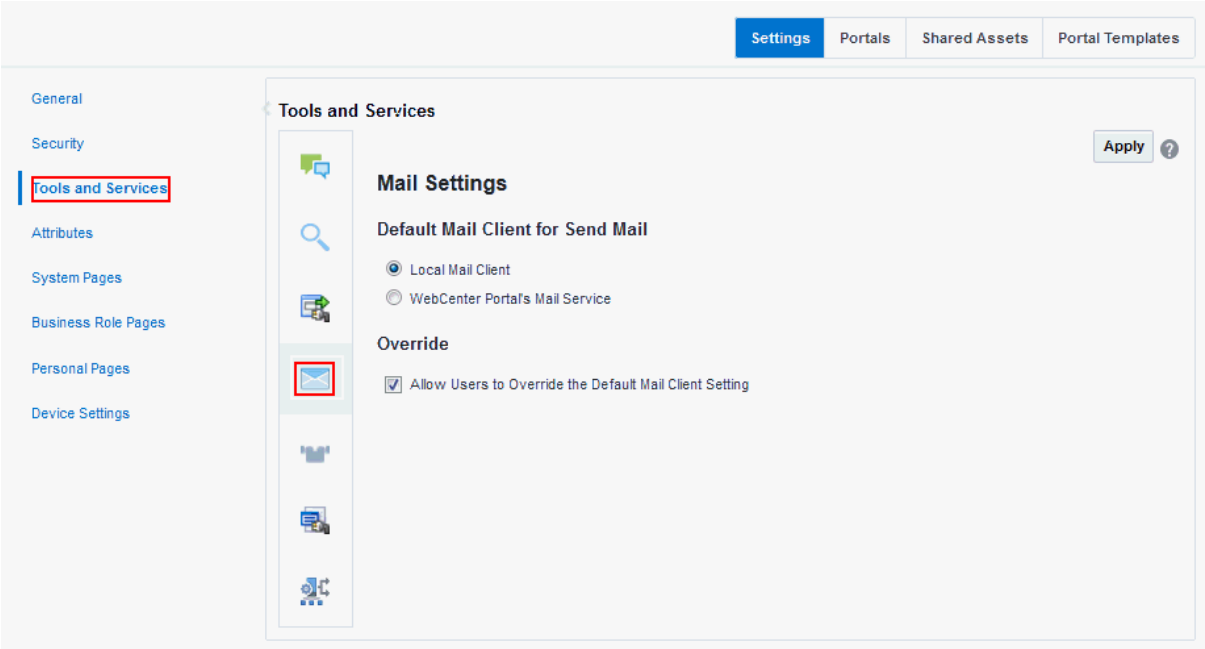
For Microsoft Exchange, go to **Administrative Tools** - **Services** to confirm that the following components are running (Status: Started):

- Microsoft Exchange IMAP4
- Simple Mail Transfer Protocol (SMTP)

Configuring Send Mail Notifications for WebCenter Portal

System administrators are responsible for setting mail options through WebCenter Portal administration settings.

Figure 12-5 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 12-6).

Figure 12-6 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 an announcement, Send Mail opens a mail window prepopulated with the title of the announcement.

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. Portal Mangers (and anyone granted the `Manage Security and Configuration` permission on the portal) set this through the Tools and Services page in the portal's administration settings. See Configuring the Mail Distribution List for a Portal in *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 managers can specify whether portal members 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.

Enabling Shared Mail Connections for Send Mail Notifications

Users do not need to specify credentials while sending mail using WebCenter Portal's Mail service when *shared credentials* are configured for the external application associated with the mail server connection.

To enable shared mail connections:

1. Ensure you have set up a mail connection that uses an external application configured with the shared credentials, and note down the mail connection name.
2. Configure WebCenter Portal to use WebCenter Portal Mail service to send mail:
   a. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Tools and Services**.

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

      `http://host:port/webcenter/portal/admin/settings/tools`

      > **See Also:**
      >
      > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

   b. Click the icon for Mail Settings.
   c. Under **Default Mail Client for Send Mail**, select **WebCenter Portal's Mail Service**.
   d. Click **Apply**.

Portal managers can now specify the name of the shared mail connection in the portals where shared mail credentials are required.

13

Managing People Connections

Configure People Connections in WebCenter Portal to create social networking tools and track portal user activities.

> **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 granted through WebCenter Portal Administration.
>
> For more information about roles and permissions, see Understanding Administrative Operations, Roles, and Tools.

Topics:

- About the People Connections Service
- People Connections Prerequisites
- Configuring People Connections for WebCenter Portal
- Archiving the Activity Stream Schema
- Specifying a Management Chain for Organization View
- Setting Profile Configuration Properties
- Synchronizing Profiles with the Identity Store

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.

People Connections features 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 information about yourself and viewing the information of other users
- **Publisher** for publishing status messages and posting files and links

The People Connections service provides task flows for using it features. For information on adding People Connections functionality to a portal, see Adding Connections to a Portal in *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. Any changes you make to WebCenter Portal post-deployment are stored in MDS metadata store as customizations. 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 where the application is deployed.

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 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 Configuring the Identity Store.

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:

- Accessing People Connections Administrative Settings
- Configuring Activity Stream
- Configuring Connections
- Configuring Profile
- Configuring Message Board
- Configuring Feedback

Accessing People Connections Administrative Settings

To access People Connections administrative settings:

1. In the portal browser, click the **Administration** tile, then click **Settings**.
2. On the **Settings** page, click **Tools and Services**.

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

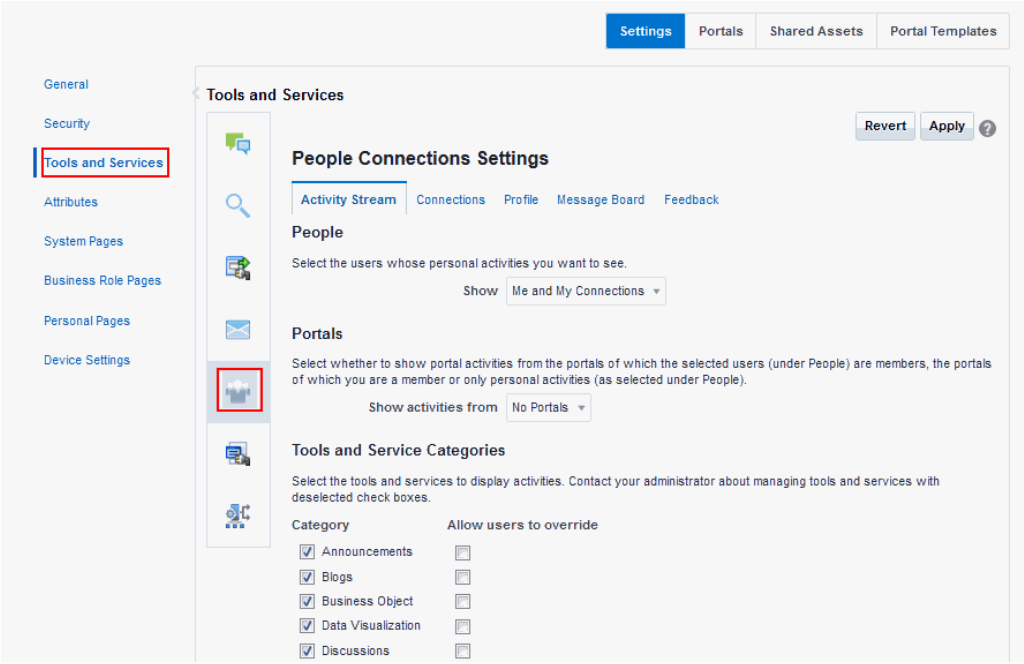
   `http://host:port/webcenter/portal/admin/settings/tools`

3. Click **People Connections**.

Configuring Activity Stream

Activity Stream is for publishing and tracking users' application activity. Activity Stream configuration settings specify the users and activities that are streamed, who can see a user's streamed activities, and whether liking and commenting is available on each streamed activity.

Figure 13-1 Administration Settings for People Connections



Who can view a user's activities and the types of activities tracked depend on Activity Stream configuration.

Table 13-1 Activities Tracked by Activity Stream

Feature Area	Tracked Activities	Scope	Activities Shared or Private
Connections	• Invitations to connect • People are connected	Home portal	Shared with inviter and invitee's connections
Documents	• Upload document from Publisher	• Portal • Home portal	• Only document uploads via Activity Stream are tracked.
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 (see Setting Feedback Preferences in *Using Portals in Oracle WebCenter Portal*).

Table 13-1 (Cont.) Activities Tracked by Activity Stream

Feature Area	Tracked Activities	Scope	Activities Shared or Private
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 (see Setting Message Preferences in *Using Portals in 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 (see Setting Profile Preferences in *Using Portals in 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 (see Configuring Activity Stream and Setting Activity Stream Preferences in *Using Portals in Oracle WebCenter Portal*).

Configure Activity Stream to show or hide actions from these 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 like a posted activity

To configure Activity Stream for all users:

1. On the **Settings** page, click **Tools and Services**.

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

`http://host:port/webcenter/portal/admin/settings/tools`

2. Click the icon for People Connections Settings.
3. Click the **Activity Stream** tab.
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 the Home portal in the current user's view of his or her Activity Stream.
5. Under **Portals**, select to show activities from:
   - **All Portals**—All portals the user has access to
   - **My Portals**—All portals the user manages
   - **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 13-1 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 13-2 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.

Table 13-2 (Cont.) Activity Stream Privacy Options

Option	Description
Allow Owner Override	Enable users to override the application default settings using their own People Connections Preferences.

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 like an Activity Stream item. Deselect the check box to prevent users from liking.

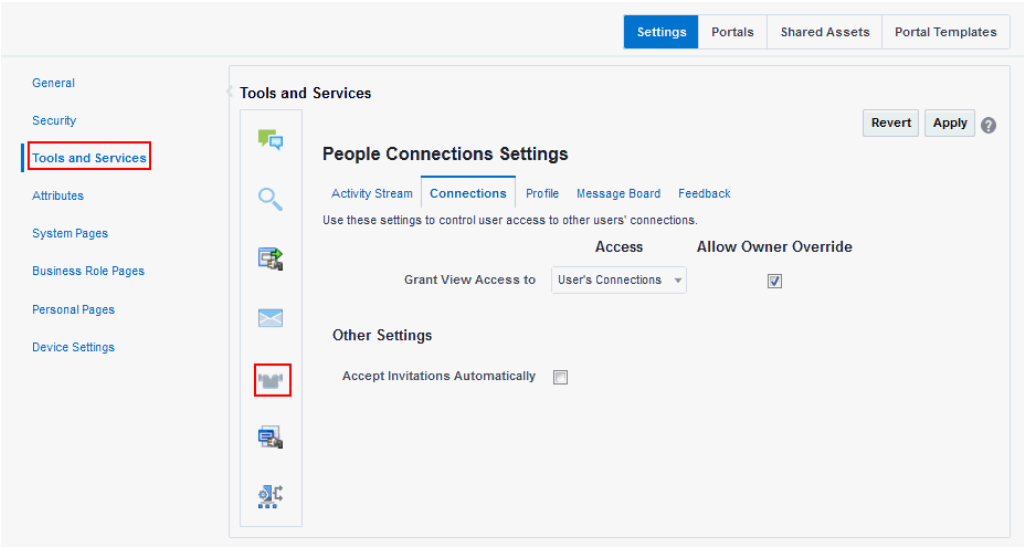
> **Tip:**
>
> Users can like and comment on streamed items that include objects. For example, users can like or comment on "Jack posted a message." Users cannot like or comment on system messages, such as "Jack and Jill are now connected."

10. Click **Apply**.

Configuring Connections

Connections configuration involves selecting who can view another user's connections and whether users accept invitations to connect automatically.

Figure 13-2 Configuration Settings for Connections



To configure Connections:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Tools and Services**.

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

   `http://host:port/webcenter/portal/admin/settings/tools`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click the icon for People Connections Settings.
3. Click the **Connections** tab.
4. Select the required connection options:

Table 13-3 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	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. • Deselect to prohibit users from overriding the administrative View access setting.
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.

5. Click **Apply**.

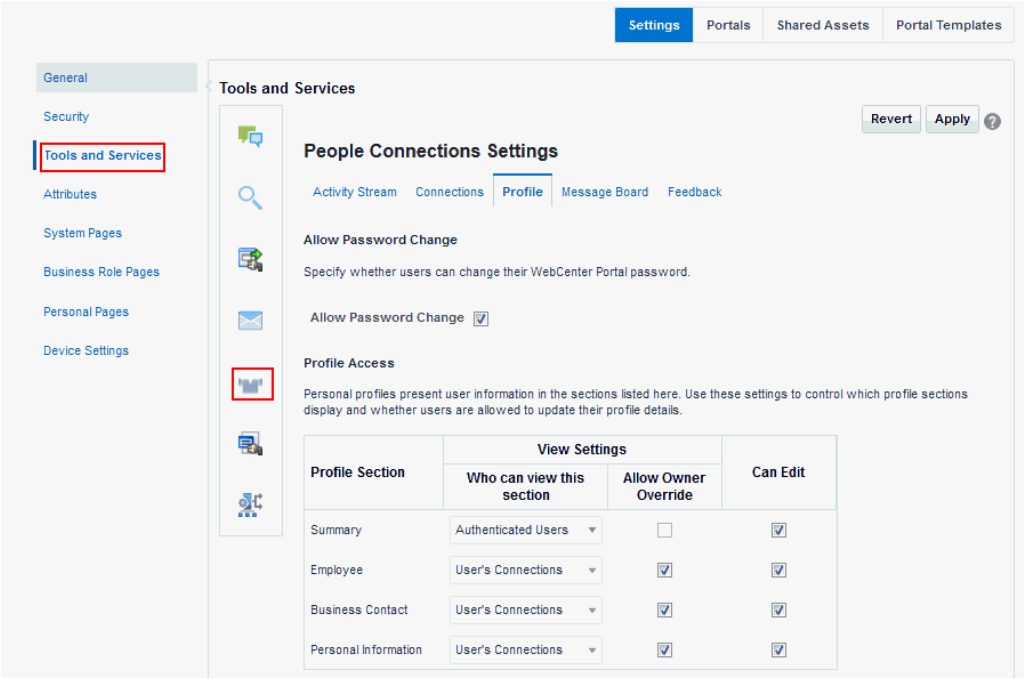
Configuring Profile

Every authenticated user has a profile that displays personal information, such as 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

WebCenter Portal. The three exceptions are: profile photo, expertise, and Publisher status messages.

Use People Connections Settings for Profile to specify whether users are allowed to change their sign in password, which profile sections display for different types of users, whether users are allowed to update their profile details, and the profile settings that users may edit.

Figure 13-3 Configuration Settings for Profile



Personal profiles are presented in four profile sections: **Summary**, **Employee**, **Business Contact**, **Personal Information**. Each section provides information related to the section name. For example, **Summary** includes a collection of basic details, such as the user's name, email address, and office location.

To configure Profile:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Tools and Services**.

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

   `http://host:port/webcenter/portal/admin/settings/tools`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click the icon for People Connections Settings.

3. Click the **Profile** tab.
4. Select the required options:

Table 13-4 Profile Configuration Options

Option	Description
Allow Password Change	Specify whether users are allowed to change their sign in 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.
Profile Access	Specify which profile sections to show and whether users are allowed to update their profile details. **View Settings**—Specify which users can view the associated profile section, and whether users can change these defaults in their personal Preferences. **Note**: **View Settings** for the **Summary** section control not only who can view summary details but also 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 selected; if **None** is selected, then the user will not appear in search results. Set values for: • **Who can view this section**—Specify which types of users can view the associated profile section by default: **Everyone**—All users, including unauthenticated (public) users, 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 profile section is hidden from all users. • **Allow Owner Override**—Select to enable users to override the default application settings you specify here in their Preferences; deselect to disable override. Users cannot change the privacy settings on the **Summary** section through their Preferences. **Can Edit**—Select to enable users to edit the associated profile section of their own personal profiles; deselect to prohibit editing. 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	Specify the profile 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.

Table 13-4 (Cont.) Profile Configuration Options

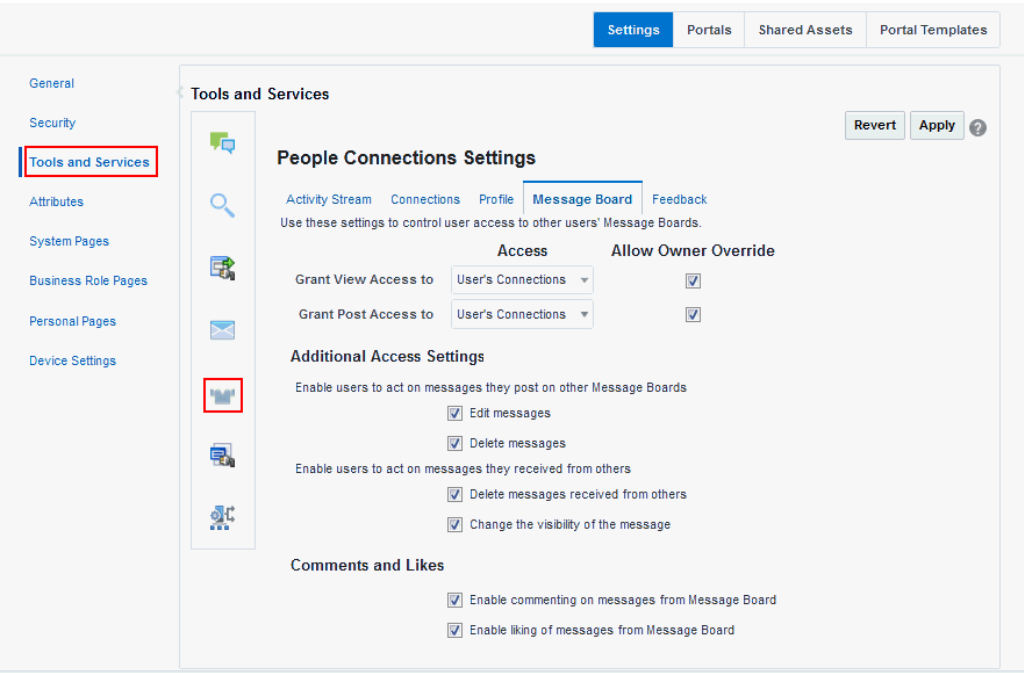
Option	Description
Profile Synchronization settings	Specify profile synchronization settings: • Specify the size of the LDAP read batch for profile synchronization. • Specify whether user profile photos will be synchronized with LDAP when the cache expires.

5. Click **Apply**.

Configuring Message Board

Message Boards provide a way for users to view and post 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 13-4 Configuration Settings for Message Board



To configure Message Board:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Tools and Services**.

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

   `http://host:port/webcenter/portal/admin/settings/tools`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click the icon for People Connections Settings.
3. Click the **Message Board** tab.
4. Specify the required options:

Table 13-5 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. • 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. • Deselect to prohibit commenting.

Table 13-5 (Cont.) Message Board Configuration Options

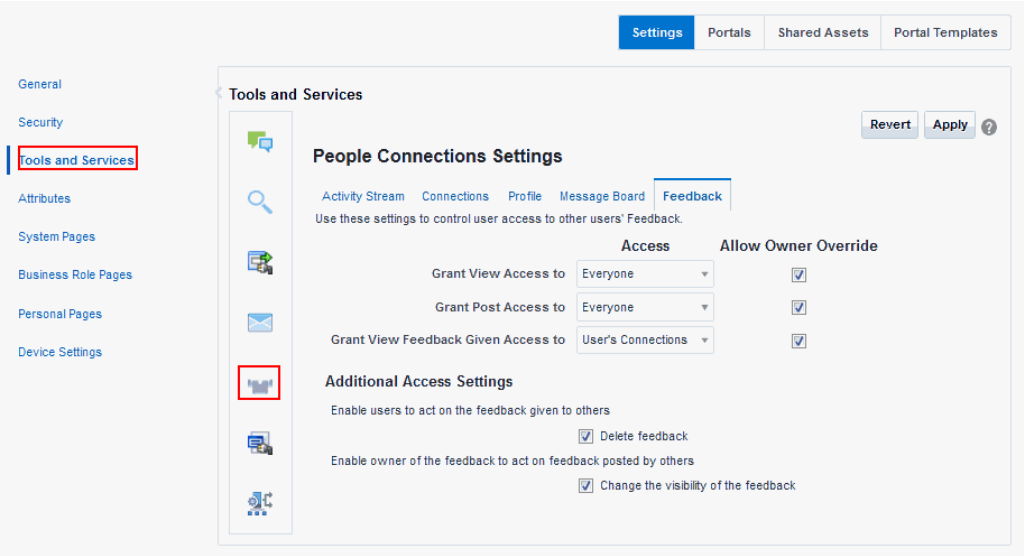
Option	Description
Enable liking of messages from Message Board	Specify whether to enable users to like a message. • Select to permit users to like messages. A **Like** link appears below each message. • Deselect to prohibit liking.

5. Click **Apply**.

Configuring Feedback

Feedback provides a way for users to view and post feedback for other application users. Configuration settings for Feedback provide controls for granting view and post access for feedback a user receives, granting view access for feedback a user gives, allowing users to override administrative default settings, enabling users to delete the feedback they post, and enabling a user to show or hide feedback left by others.

Figure 13-5 Configuration Settings for Feedback



To configure Feedback:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Tools and Services**.

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

   `http://host:port/webcenter/portal/admin/settings/tools`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click the icon for People Connections Settings.
3. Click the **Feedback** tab.
4. Select the required options:

Table 13-6 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. • 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.

5. Click **Apply**.

Archiving the Activity Stream Schema

Administrators can use these WLST commands to archive and restore data in the Activity Stream schema:

- `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`—Truncate 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 Activity Stream in *WebCenter WLST Command Reference Reference*.

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, for example, they can render a view of a manager and the manager's direct reports.

Figure 13-6 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.

> **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 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.

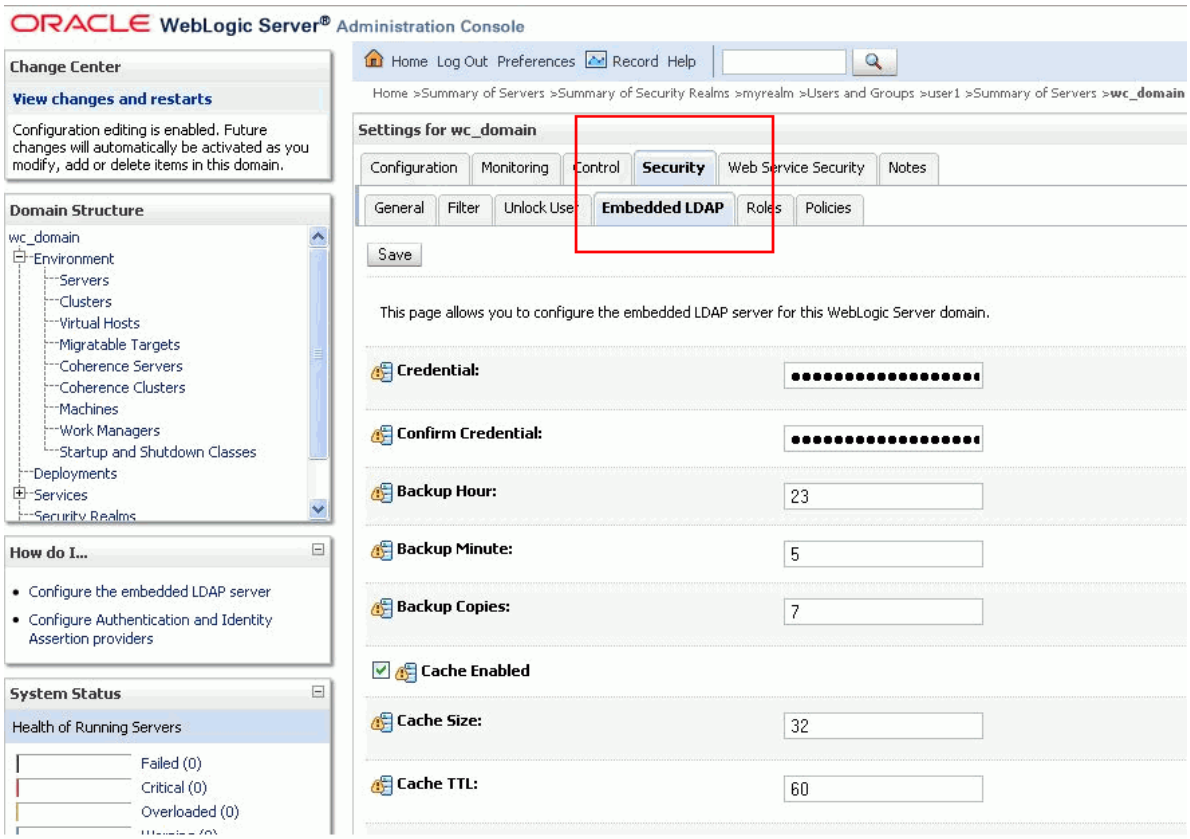
In this example, there are three users:

- `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** subtab (Figure 13-7).

Figure 13-7 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.

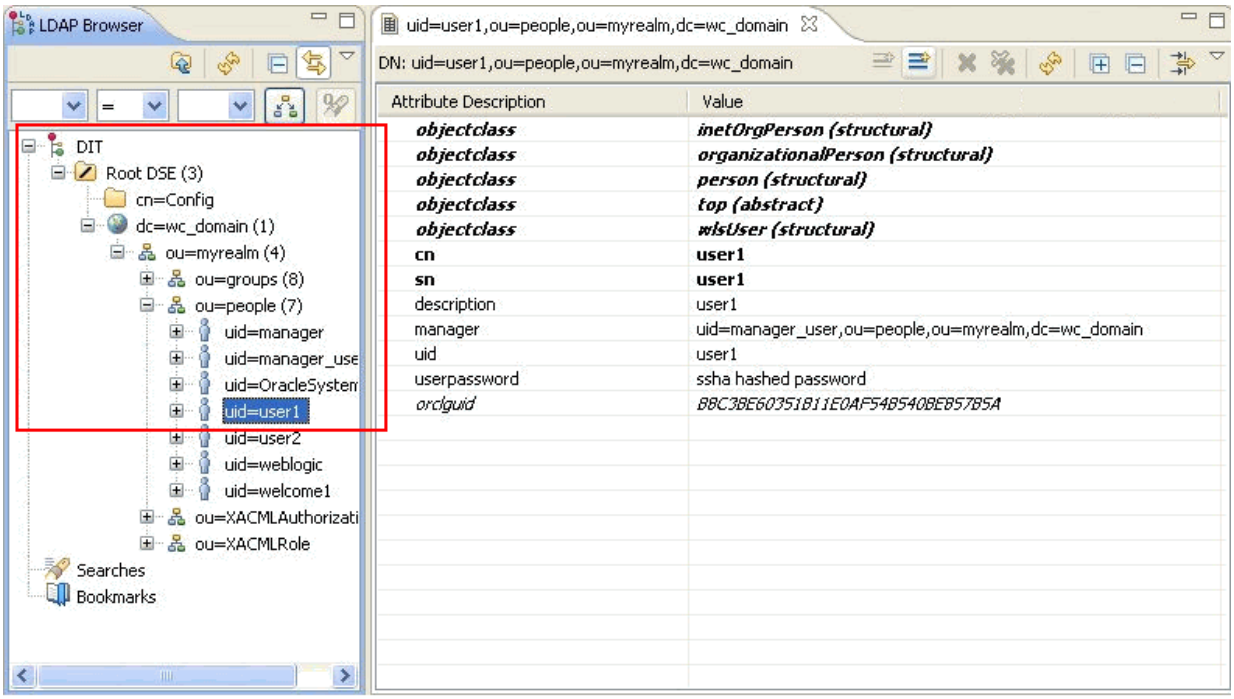
d. Restart your administration and managed servers.

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 13-8).

   For example, click in succession:
   - `dc=wc_domain`
   - `ou=myrealm`

- ou=people
- uid=user1

Figure 13-8 Selecting a User in the DIT Tree of an LDAP Browser



4. In the **Attribute Description** column, add a new attribute of type `manager`.

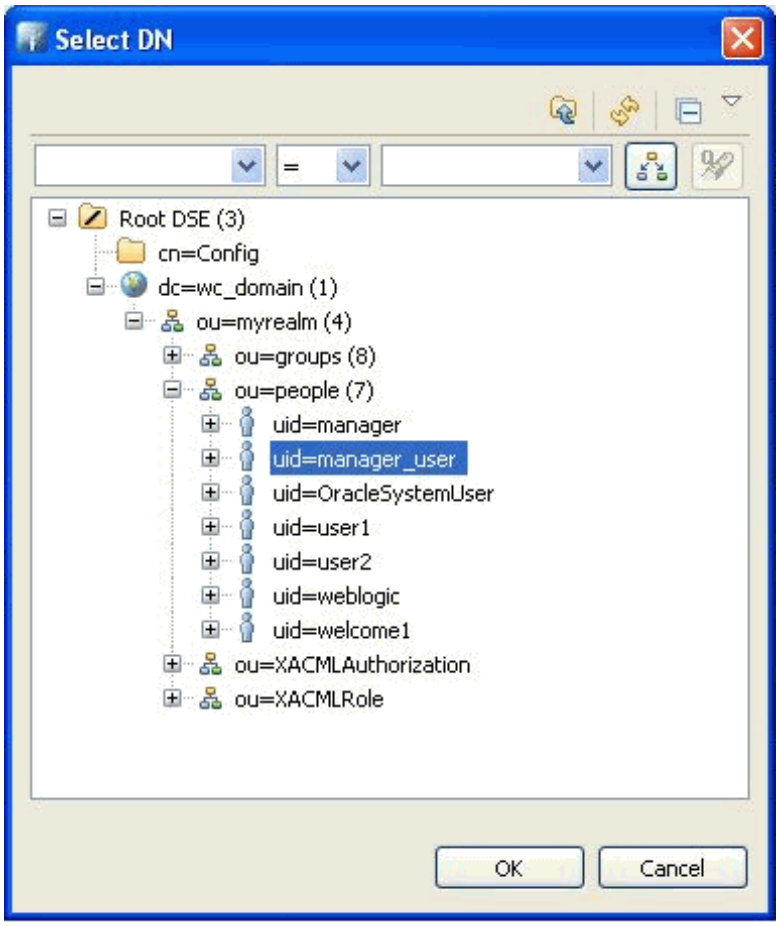
> **Tip:**
>
> Press Ctrl-Shift-+ to open the New Attribute dialog.

5. For the attribute value, select the DN for `manager_user` (Figure 13-9).

   For example, under the root, select in succession:

   - dc=wc_domain
   - ou=myrealm
   - ou=people
   - uid=manager_user

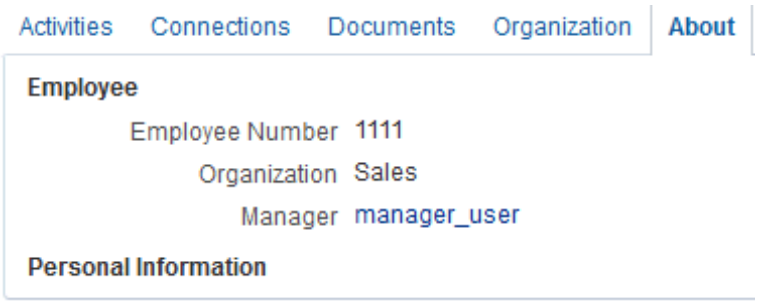
Figure 13-9 Select DN Dialog



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 shown as the manager (Figure 13-10).

Figure 13-10 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.

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:

  ```
  setProfileConfig(appName, [ProfilePageVersion], [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 where the WebCenter Portal application is deployed.

  - `ProfileSyncHourOfDay` - (Optional) 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 to perform this operation on. 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 to perform this operation on. For example, `webcenter`.
  - `key` - Name of a the Profile Config property to get. Valid values include:
    - `ProfilePageVersion`
    - `ProfileSyncHourOfDay`
    - `ProfileSyncFrequencyInDays`
  - `server` - (Optional) The name of the target server where the application is deployed.
  - `applicationVersion` - (Optional) The version number of the application.

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 `setProfilePhotoSync`.
  - Synchronize profile information for a specific user by running `syncProfile`.

For more information, see the following command references in *WebCenter WLST Command Reference Reference*:

- startSyncProfiles
- stopSyncProfiles
- isSyncProfilesRunning
- setProfilePhotoSync
- syncProfile

14

Managing RSS

Configure and manage RSS functionality for WebCenter Portal.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

> **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 granted through WebCenter Portal Administration.
>
> For more information about roles and permissions, see Understanding Administrative Operations, Roles, and Tools.

Topics:

- About RSS
- RSS Prerequisites
- Setting Up a Proxy Server for External RSS News Feeds
- Testing External RSS News Feed Connections

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. RSS also delivers content update information from various portal resources including discussions, lists, and announcements.

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 to display content from external RSS news feeds.

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. 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:

- Setting Up a Proxy Server Using Fusion Middleware Control
- Setting Up a Proxy Server Using WLST

Setting Up a Proxy Server for External RSS News Feeds

To enable external RSS news feeds in WebCenter Portal, 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.

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 WebCenter Portal.
2. From the **WebCenter Portal** menu, select **Settings** > **Application Configuration**.
3. In the **Proxy Server** section, enter the host name and the port number of the proxy server. For details, see Table 14-1.

Table 14-1 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.

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 setWebCenterProxyConfig in *WebCenter WLST Command Reference Reference*.

For information about how to run WLST commands, see 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.

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')
```

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 test the RSS connection, you need to add the RSS task flow to a portal page and set the URL to an external RSS feed. If the RSS feed displays correctly, proxy configuration is set up properly. For information about adding the RSS task flow and editing the URL, see Adding RSS News Feeds to a Portal in *Building Portals with Oracle WebCenter Portal*.

15

Managing Subscriptions and Notifications

Administer subscriptions and notifications by creating and enforcing application-wide defaults for application-level subscriptions, specifying the server to handle notification delivery, and using WLST commands to set and get notification messaging configuration details.

> **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. In WebCenter Portal, the `Administrator` role is granted through WebCenter Portal Administration.
>
> For more information about roles and permissions, see Understanding Administrative Operations, Roles, and Tools.

Topics:

- About Subscriptions and Notifications
- Setting Up Default Subscription Preferences
- Setting Up Notifications
- Creating and Applying Custom Notification Templates
- Testing the Notifications Connection

About Subscriptions and Notifications

In WebCenter Portal, subscriptions and notifications provide users with a way to subscribe to the types of services and application objects that interest them. Consequently, users receive timely notice over their selected messaging channels of changes that affect their subscribed services and objects.

Always use the Fusion Middleware Control or WLST command-line tool to review and configure back-end services for WebCenter Portal. Any changes you make to WebCenter Portal, post deployment, are stored in the MDS metadata store as customizations.

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 Also:**
>
> For information about adding notifications functionality to a portal, see Adding Notifications to a Portal in *Building Portals with Oracle WebCenter Portal*.

Setting Up Default Subscription Preferences

WebCenter Portal users set their personal Subscriptions preferences through the WebCenter Portal 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:

- About Subscription Defaults
- Setting Subscription Defaults
- Setting Subscriptions Preferences in WebCenter Portal

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 WebCenter Portal (for more information, see Subscribing to the Application, to Portals, and to Objects in *Using Portals in Oracle WebCenter Portal*.)

Each setting provides three attributes:

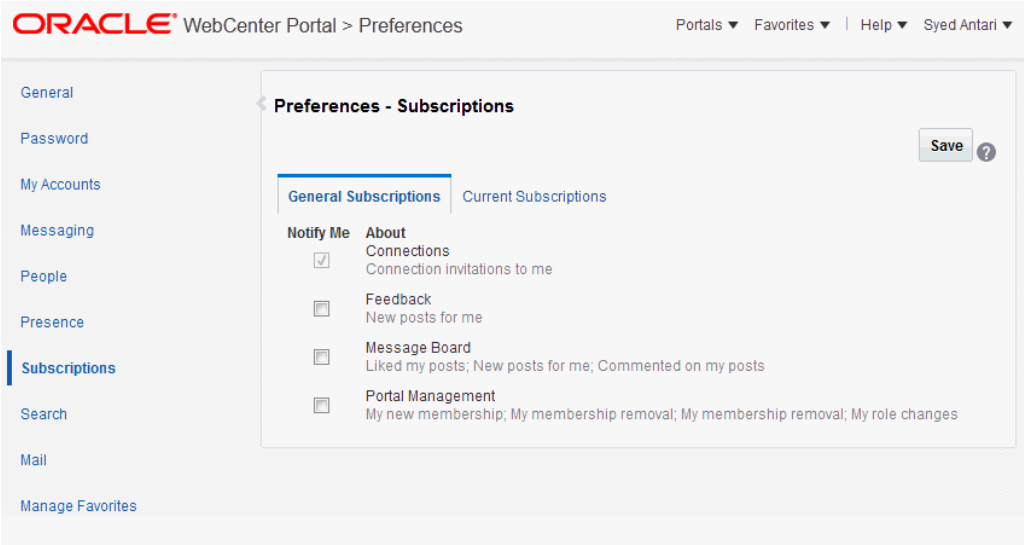
- `id`—for specifying the service ID:
  - `oracle.webcenter.peopleconnections.connections`, the Connections feature of the People Connections service
  - `oracle.webcenter.peopleconnections.wall`, the Message Board feature of the People Connections service
  - `oracle.webcenter.peopleconnections.kudos`, the Feedback feature of the People Connections service
  - `oracle.connections. community`, portal membership management
- `subscription-enabled`—for specifying the initial state of the preference option: `true` (enabled) or `false`(not enabled)

> **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** page in Preferences.

Figure 15-1 Preferences - Subscriptions: General Subscriptions Page



The following table illustrates the effect of custom administrator-level subscriptions settings on the appearance of the **General Subscriptions** tab.

Table 15-1 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 15-1, the most typical scenario for most notifications is `false/true` (row 3).

The following table lists the types of actions that can trigger an application-level notification and associates them with their related service ID.

Table 15-2 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 *Portal Manager* to another custom role	`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`

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.

   The following example provides sample content for an application-wide subscription preferences setting file and an example of each required option.

```
<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.

   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 where 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`.

   For information about the `importMetadata` WLST command, see importMetadata in *WLST Command Reference for Infrastructure Components*.

The table describes the effect of various combinations of settings for the service ID `oracle.webcenter.peopleconnections.connections`.

Table 15-3 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

The table describes the effect of various combinations of settings for the service ID `oracle.webcenter.peopleconnections.wall`.

Table 15-4 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.

The table describes the effect of various combinations of settings for the service ID `oracle.webcenter.peopleconnections.kudos`.

Table 15-5 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.

Table 15-5 (Cont.) Effect of Subscription Configurations for Feedback

subscription-enabled	end-user-configurable	Effect
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.

The table describes the effect of various combinations of settings for the service ID `oracle.webcenter.community`.

Table 15-6 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.

Setting Subscriptions Preferences in WebCenter Portal

Individual users set their own subscription preferences in 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 Subscribing to the Application, to Portals, and to Objects in *Using Portals in Oracle WebCenter Portal*.

- **Messaging**, where users access controls for configuring their preferred messaging channels and filters (BPEL connection types only)

For more information, see Establishing and Managing Your Messaging Channels and Filters in *Using Portals in Oracle WebCenter Portal*.

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:

- About Connection Channels
- Notification Prerequisites
- Configuration Roadmap for Notifications
- Specifying the Notifications Channel Using Fusion Middleware Control
- Specifying the Notifications Channel Using WLST
- Example - Setting Up Mail Notifications for WebCenter Portal Using WLST

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 two messaging channel options to users: mail and texting (SMS).
- **Mail Server** delivers notification messages exclusively through a mail server that is configured for WebCenter Portal.

BPEL Server Connection Type

Selection of a BPEL server presupposes that you have established a connection to a BPEL server with the User Messaging Service (UMS) is available. For information about connecting to a BPEL server, see Managing the SOA Connection for WebCenter Portal Membership Workflows.

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 for Notifications that you use for `setSpacesWorkFlowConnectionName`, provided you use the BPEL Server for notifications.

Mail Server Connection Type

Selection of a mail server presupposes that you have established a connection to 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 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 to open User Messaging Preferences, any changes you make are ignored.

> **See Also:**
>
> Establishing and Managing Your Messaging Channels and Filters in *Using Portals in Oracle WebCenter Portal*

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:

- Installation
- Configuration
- Security
- Limitations

Installation

Installation requirements associated with Notifications change according to the type of connection you select for Notifications messaging.

If you will 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 messaging channels, you must install drivers for these as well.

If you will use the Mail service for Notifications messaging, no Notifications-specific installation is required, but the Mail service must be configured as described in Managing Mail.

Configuration

Configuration prerequisites for Notifications also depend on the connection type you select for Notifications messaging.

BPEL Server Configuration

If you want users to have messaging channel options—mail and texting (SMS)—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 mail driver is configured. The SMS driver is available, but must be deployed.

Mail Server Configuration

If you want 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.

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.

Security

There are no security considerations specifically associated with Notifications.

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 and SMS.

In WebCenter Portal 12*c*, Content Manager, a new Documents service task flow, has been introduced. Integration between the Content Manager task flow and subscriptions and notifications is not in place; consequently, no document-related activities trigger notifications.

Configuration Roadmap for Notifications

The section provides an overview of the prerequisites and tasks required to get the Notifications service working in WebCenter Portal.

Figure 15-2 Configuring the Notifications Service

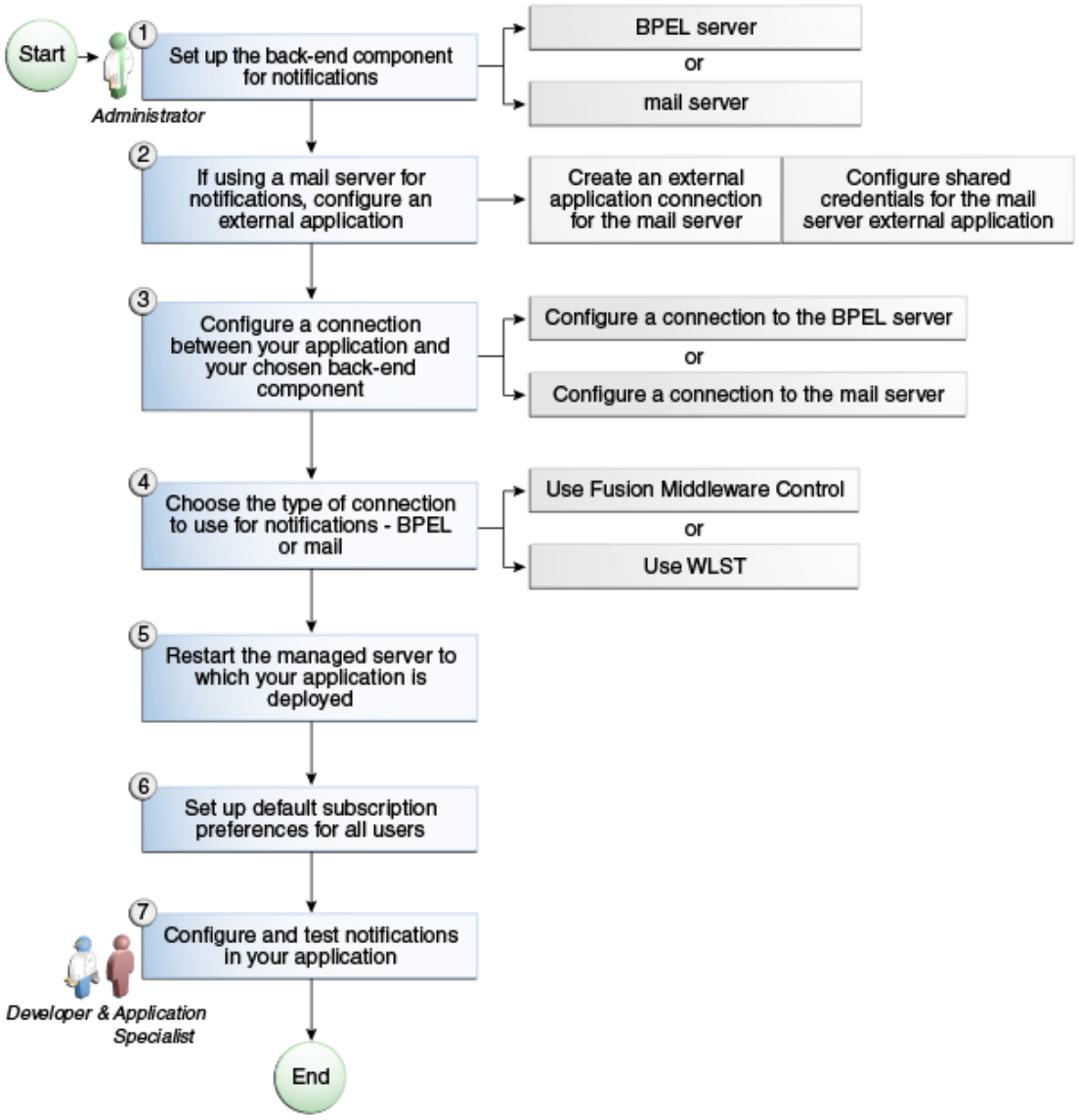


Table 15-7 Configuring Notifications

Actor	Task	Link
Administrator	**1.** Set up one of the following back-end componentsfor Notifications. • Set up the BPEL server • Set up the mail server	• BPEL Server Prerequisites • Mail Server Prerequisites
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	Registering External Applications at Runtime

Table 15-7 (Cont.) Configuring Notifications

Actor	Task	Link
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	• Specifying the BPEL Server Hosting WebCenter Portal Workflows • Registering Mail Servers
Administrator	**4.** Choose the type of connection to use for Notifications, either BPEL or Mail, using one of the following tools: • Fusion Middleware Control • WLST	• Specifying the Notifications Channel Using Fusion Middleware Control • Specifying the Notifications Channel Using WLST
Administrator	**5.** Restart the managed server (`WC_Portal`) where WebCenter Portal is deployed.	Starting and Stopping the WebCenter Portal Application
Administrator	**6.** Set up default subscription preferences for all users	Setting Up Default Subscription Preferences
Application Specialist/End User	**7.** Configure and test Notifications in WebCenter Portal as an: • application specialist • end user	• Adding Notifications to a Portal in *Building Portals with Oracle WebCenter Portal* • Getting Notified When Things Change in *Using Portals in Oracle WebCenter Portal*

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.
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. To make your changes take effect, restart the managed server where WebCenter Portal is deployed.

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 setNotificationsConfig in *WebCenter WLST Command Reference Reference*. See also getNotificationsConfig in the same guide.

> **Note:**
>
> Updates to this configuration are stored in the MDS repository. For configuration changes to take effect, you must restart the managed server where the application is deployed.

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`.

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 mail account from which mail notifications will be sent. This must be in the format `<user>@<domain of the mail server>`.

   Optionally, you may add the following fields to use while sending out the mail notification.

   ```
   addExtAppField(appName='webcenter',name='NotificationSharedApp',fieldName='Email
   Address',fieldValue='sender's_email_address',displayToUser=false)
   addExtAppField(appName='webcenter',name='NotificationSharedApp',fieldName='Your
   Name',fieldValue='sender's_display_name',displayToUser=false)
   ```

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`.

5. Set Mail as the notifications channel:

```
setNotificationsConfig(appName='webcenter', type='MAIL',
                       name='NotificationSharedConn')
```

This sets `NotificationSharedConn` as the mail connection to use when sending notifications.

6. For the changes to take effect, restart the managed server where WebCenter Portal is deployed (`WC_Portal` 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 `Portal Manager` to another custom role to trigger a notification.

For more information about WLST commands, see WebCenter Portal Custom WLST Commands in *WebCenter WLST Command Reference Reference*.

Creating and Applying Custom Notification Templates

The notification messages that users receive through 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:

- About Overwriting Default Notification Templates
- Overwriting a Default Notifications Template

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` (or `defaultTemplate_rtl.xml`—when right-to-left language support is required).

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.

The following example illustrates the default content of notification message template files. You can use this to formulate your custom files.

> **Note:**
>
> Differences may appear between custom files particularly under the <style> tag, where alignment—either right or left—is specified.

Example

```
<?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>

    <?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>
```

Overwriting a Default Notifications Template

To overwrite a default notifications template (an xml file) to customize notification message formats:

1. Create a directory in the format of: `/tmp/repository/oracle/webcenter/notification/custom/template`

   This will later be used to import the files into the MDS and override the original application file.
2. Inside the directory you created, create a custom XML file with the name `defaultTemplate.xml` (or `defaultTemplate_rtl.xml`, for a right-to-left language template).
3. Populate the custom file with your revised version of one of these default files.
4. Upload the custom file into WebCenter Portal's MDS repository using the `importMetadata()` WLST command. Overwrite the original file, placing the custom file where the absolute path to the file contains the namespace `oracle/webcenter/notification/custom`.

For example:

```
importMetadata(application='webcenter', server='WC_Portal',
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, inside the MDS repository. For example, for a file created under the following namespace:

```
/tmp/repository/oracle/webcenter/notification/custom/template/
defaultTemplate.xml
```

5. Upload the custom file into WebCenter Portal's MDS repository by running the `importMetadata()` WLST command.

   For example:

   ```
   importMetadata(application='webcenter', server='WC_Portal',
        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, for a file that is created under the following namespace:

   ```
   /tmp/repository/oracle/webcenter/notification/custom/template/
   defaultTemplate.xml
   ```

   The `fromLocation` is `/tmp/repository` since the remaining sub-directory consists of the namespace for the XML file. The namespace must have at least the path `/oracle/webcenter/notification/custom`.

6. Restart WebCenter Portal.

> **Note:**
>
> For information about the `importMetada` and `exportMetada` WLST commands, see Application Metadata Management Commands in *WLST Command Reference for Infrastructure Components*.

Testing the Notifications Connection

In general, Notifications is dependent on the underlying Mail or BPEL connection being 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 Testing Mail Server Connections.

16

Managing the SOA Connection for WebCenter Portal Membership Workflows

Configure and manage the SOA connection for membership workflow notifications from WebCenter Portal to appear in Oracle BPM Worklist.
Always use Fusion Middleware Control or WLST command-line tool to review and configure back-end servers for WebCenter Portal. Any changes that you make to WebCenter Portal post-deployment are stored in MDS metadata store as customizations.

> **Note:**
>
> Changes that you make to the SOA connection 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 Starting and Stopping Managed Servers for WebCenter Portal Application Deployments.

> **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 WebCenter Portal Administration.
>
> For more information about roles and permissions, see Understanding Administrative Operations, Roles, and Tools.
>
> For troubleshooting issues with BPM worklists, see Email Notifications Not Working

Topics:

- Configuration Roadmap for WebCenter Portal Workflows
- About BPEL Connections
- BPEL Server Prerequisites
- Specifying the BPEL Server Hosting WebCenter Portal Workflows
- Configuring WebCenter Portal Workflow Notifications to be Sent by Email
- Excluding Webcenter Portal Workflows URL in OAM

Configuration Roadmap for WebCenter Portal Workflows

Table 16-1 in this section provides an overview of the prerequisites and tasks required to use Oracle BPM Worklist in WebCenter Portal.

Table 16-1 Configuring Workflows for WebCenter Portal

Actor	Task	Link
Administrator	**1.** Install WebCenter Portal and Oracle SOA Suite	• Preparing to Install and Configure Oracle WebCenter Portal in *Installing and Configuring Oracle WebCenter Portal* • Preparing to Install and Configure Oracle SOA Suite and Oracle Business Process Management in *Installing and Configuring Oracle SOA Suite and Business Process Management*
Administrator	**2.** Create the BPEL connection and enable the WebCenter Portal Workflow using either: • Fusion Middleware Control • WLST: Use `createBPELConnection` to create the BPEL connection and `setSpacesWorkflowConnectionName` to enable the workflow	• Fusion Middleware Control: Specifying the BPEL Server Hosting WebCenter Portal Workflows. • WLST: createBPELConnection and setBPELConnection in *WebCenter WLST Command Reference Reference*.
Administrator	**3.** Deploy the WebCenter Portal workflows: • `sca_CommunityWorkflows.jar`, which is available at `/Oracle_Home/wcportal/common/soa-composite/wcp/` • `WebCenterWorklistDetailApp.ear`, which is available at `Oracle_Home/wcportal/webcenter/applications/WebCenterWorklistDetailApp.ear` **Note:** During the SOA installation, if you select the option **Oracle Webcenter Portal Composites - 12.2.1.0 [wcportal]** on the Templates page, `WebCenterWorklistDetailApp.ear` gets deployed.	Deploying and Managing SOA Composite Applications in *Administering Oracle SOA Suite and Oracle Business Process Management Suite*
Administrator	**4.** Configure the BPEL server to use the same identity store as WebCenter Portal	Configuring the Identity Store

Table 16-1 (Cont.) Configuring Workflows for WebCenter Portal

Actor	Task	Link
Administrator	**5.** Secure the connection to the BPEL server • **5.a** (Optional) Configure Single Sign-On If using OAM R2 or later, exclude Webcenter Portal Workflows URL • **5.b** Configure WS-Security • **5.c** (Optional) Configure SSL	• **5.a** Configuring Single Sign-On If using OAM R2 or later:Excluding Webcenter Portal Workflows URL in OAM • **5.b**Configuring Web Services Security • **5.c**Configuring SSL
End User	**6.** Test that the integration to the Oracle BPM Worklist application is working in WebCenter Portal • **6.a** In WebCenter Portal, create a portal and invite members. • **6.b** Log in to Oracle BPM Worklist and view the worklist items. • **6.c** Select the invitation worklist item and click **Accept**. • **6.d** Log out and then log in as the portal manager and view the **Members** tab. Confirm that the user is now a member in the selected role and the status is not set to Invited anymore.	• **6.a** Managing Members and Assigning Roles in a Portal in *Building Portals with Oracle WebCenter Portal*. • **6.c** Using Oracle BPM Worklist in *Developing SOA Applications with Oracle SOA Suite* • **6.d** Inviting a Registered User in *Building Portals with Oracle WebCenter Portal*.

About BPEL Connections

Consider the following while working with BPEL connections:

- By configuring a default BPEL server, WebCenter Portal users can manage memberships through notifications that can be viewed in the SOA BPM worklist. For more information, see Using Oracle BPM Worklist in *Developing SOA Applications with Oracle SOA Suite*.
- WebCenter Portal workflows require a single connection to the BPEL server included with the Oracle SOA Suite. For more information, see Specifying the BPEL Server Hosting WebCenter Portal Workflows.

BPEL Server Prerequisites

Consider the following to ensure smooth functioning of worklists:

- Make sure that the Oracle BPM Worklist application is part of the SOA server. The URL is in the following format:

  `http://host:port/integration/worklistapp`

  If Oracle BPM Worklist 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 WebCenter 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 Oracle BPM 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 Enabling the weblogic User for Logging in to the Worklist in *Developing SOA Applications with Oracle SOA Suite*
- 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 Troubleshooting WebCenter Portal Workflows.

This section includes the following subsections:

- BPEL Server - Installation and Configuration
- BPEL Server - Security Considerations

BPEL Server - Installation and Configuration

WebCenter Portal uses the BPM Worklists 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 Preparing to Install and Configure Oracle SOA Suite and Oracle Business Process Management in *Installing and Configuring Oracle SOA Suite and Business Process Management*.

After installing Oracle SOA Suite, you can configure WebCenter Portal to use the BPEL server for viewing and managing worklists.

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 WebCenter Portal.

For example, if the user `rsmith` wants 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 worklist task details sent from WebCenter Portal, without incurring additional login prompts, WebCenter Portal and Oracle SOA Suite servers must be configured to a shared Oracle Single Sign-On server.

For a secure connection you can configure WS-Security between SOA and WebCenter Portal.

Specifying the BPEL Server Hosting WebCenter Portal Workflows

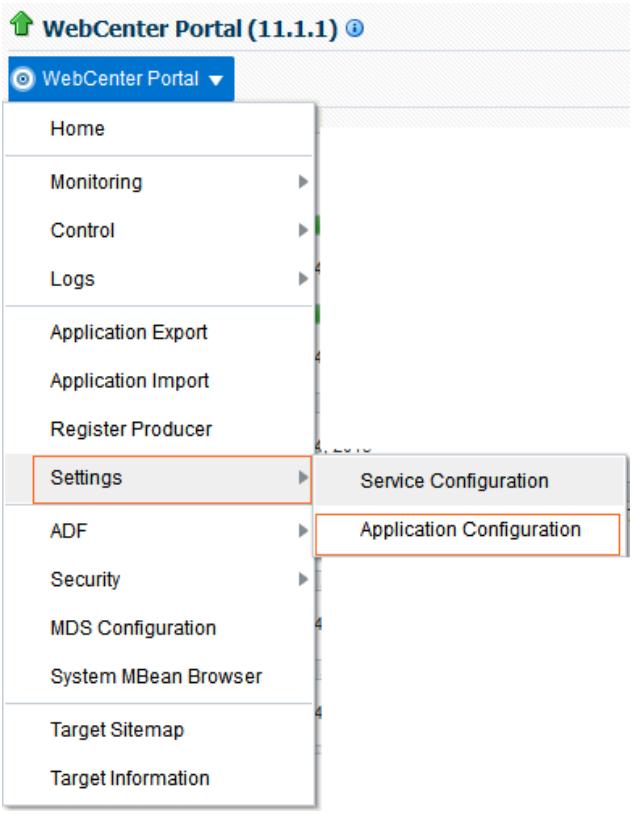
WebCenter Portal uses the BPEL server included with the Oracle SOA Suite to host internal workflows, such as worklists, portal membership notifications, portal subscription requests, and so on. To enable workflow functionality for WebCenter Portal, a connection to this BPEL server is required.

> **Note:**
>
> WebCenter Portal workflows must be deployed on the SOA managed server that WebCenter Portal is configured to use. Also, keep the following points in mind:
>
> - WebCenter and BPEL should be in same domain.
> - Both WebCenter and BPEL should be front ended with OHS.
> - The context path for WebCenter and BPEL should be same.

To configure a connection for worklist notifications:

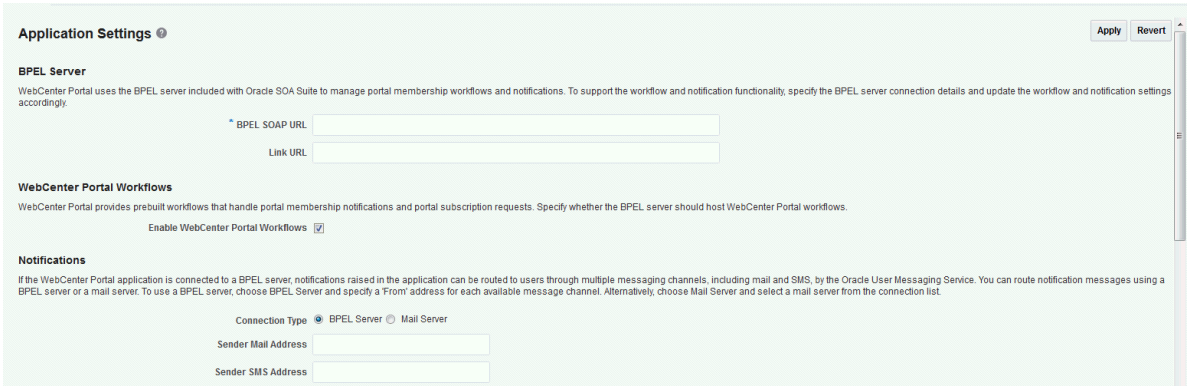
1. Log in to Fusion Middleware Control, and navigate to the home page for WebCenter Portal.
2. From the **WebCenter Portal** menu, select **Settings**, then **Application Configuration**.

Figure 16-1 WebCenter Portal Application Configuration Menu



The Application Settings page opens.

Figure 16-2 Choosing the BPEL Server Where Workflows are Deployed



3. In the BPEL SOAP URL field, specify the name of the SOA server for worklists.

   The SOA server name that you specify here will contain the BPM worklists for WebCenter Portal.

4. Select **Enable WebCenter Portal Workflows**.

5. Click **Apply**.

6. Restart WC_Portal, the managed server on which the WebCenter Portal application is deployed, to effect this change.

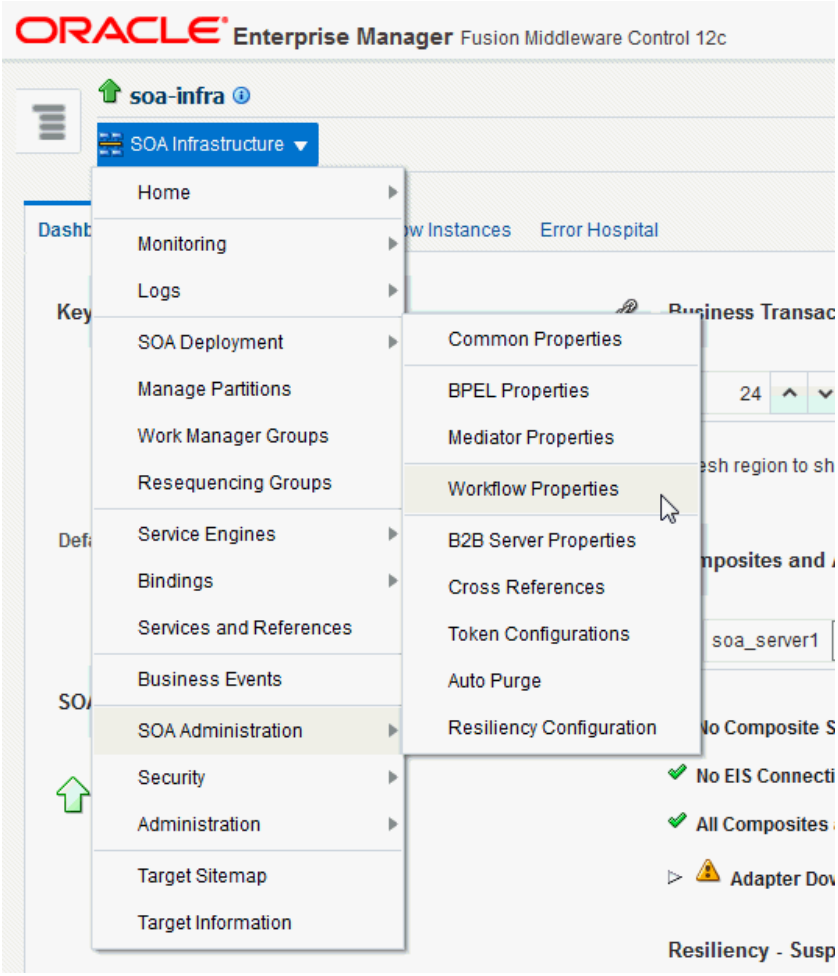
Configuring WebCenter Portal Workflow Notifications to be Sent by Email

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 briefly explains how to enable email notifications and configure your mail server details to have WebCenter Portal workflow notifications sent to users by email. For a more detailed description, see Configuring Human Workflow Notification Properties in *Administering Oracle SOA Suite and Oracle Business Process Management Suite*.

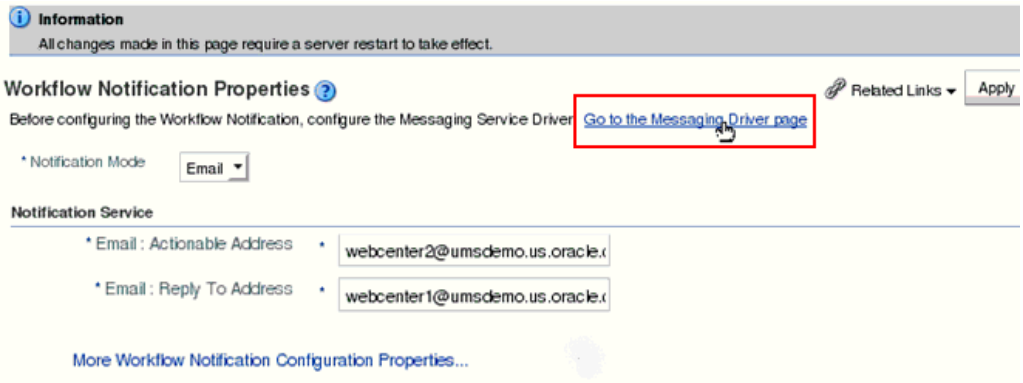
1. Use Fusion Middleware Control to update SOA to enable email notifications. Under the SOA server, select **SOA Administration**, then **Workflow Properties**, as shown in the figure.

Figure 16-3 SOA Administration - Workflow Config



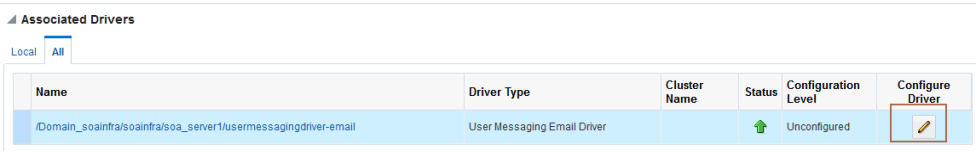
2. With **Email** selected as the **Notification Mode**, provide valid email accounts to use.

Figure 16-4 Email Notification Mode Properties



3. Click **Go to the Messaging Driver page**.
4. Select the **Configure Driver** icon for your User Messaging Email Driver.

Figure 16-5 Associated Drivers



5. To configure your email driver for notifications, see Configuring an Email Driver for Notifications in *Using Oracle Managed File Transfer*.
6. After you finish, 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.

> **Note:**
>
> To test notifications, in the portal administration settings **Members** page, you can add people and edit email notification messages. For more information, see Managing Members and Assigning Roles in a Portal in *Building Portals with Oracle WebCenter Portal*.

Excluding Webcenter Portal Workflows URL in OAM

When the Oracle SOA Suite BPEL server is configured to use OAM 11*g* R2 or later, WebCenter Portal Workflows endpoints need to be excluded from the OAM server.

To exclude WebCenter Portal Workflows endpoints in OAM 11*g* R2 or later:

1. Open the OAM Admin Console.
2. Navigate to your application domain.
3. Open the **Resources** tab, and click **Create**.

4. Create a resource of the type HTTP.
5. For **Resource URL**, enter:

   ```
   /soa-infra/services/default/CommunityWorkflows/**
   ```

6. Set the Protection Level to **Excluded**.
7. Create another HTTP type resource and specify the **Resource URL** as:

   ```
   /soa-infra/services/default/CommunityWorkflows*
   ```

8. Set the Protection Level to **Excluded**.
9. Click **Apply**.
10. Restart OHS.

17

Managing Portlet Producers

Register a WSRP portlet producer so that its portlets can be consumed in WebCenter Portal, and deploy WSRP portlet producer applications.

> **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 WebCenter Portal granted through WebCenter Portal Administration.
>
> For more information about roles and permissions, see Understanding Administrative Operations, Roles, and Tools.

Topics:

- About Portlet Producers
- Registering WSRP Producers
- Testing WSRP Producer Connections
- Editing WSRP Producer Registration Details
- Deregistering WSRP Portlet Producers
- Deploying Portlet Producer Applications
- Managing Oracle PDK-Java Portlet Producers

About Portlet Producers

WebCenter Portal enables you to expose functionality from other applications in your portals by consuming portlets provided by those applications. A portal can consume portlets provided by a third party, such as a packaged-application vendor, as well as those that are built using WebCenter Portal or other Oracle products.

Table 17-1 lists some of the products supported as portlet producers within WebCenter Portal.

By default, users with the `Administrator` role have the `AppConnectionManager` role; and therefore, application administrators can configure Portlet Producers through the WebCenter Portal Administration Console. For more information on `AppConnectionManager` role, see Default Application Roles.

Table 17-1 Supported Portlet Producers

Portlet Producer	Supported?	Notes
Oracle WebLogic Portal	Yes	For more information, see Exporting Java Portlets for Use on Other Systems in *Portlet Development Guide for Oracle WebLogic Portal*.
Oracle WebCenter Interaction	N/A	
E-Business Suite application	Yes	For more information, see Integrating E-Business Suite Applications
Peoplesoft application	Yes	For more information, see Integrating PeopleSoft Applications.
JD Edwards application	Yes	For more information, see Integrating JD Edwards Applications.

Note:

WSRP producers built by a third party and consumed by WebCenter Portal should function correctly provided:

- The producer does not rely on any vendor-specific extensions to WSRP.
- The portlets do not make assumptions about the application in which they are consumed, for example by expecting a particular JavaScript method to exist in the page.

Application administrators can register and manage portlet producers at runtime through out-of-the-box administration pages or from any page that includes the Portlet Producer task flow.

System administrators can use Fusion Middleware Control or the WLST command-line tool to register and manage portlet producers for WebCenter Portal.

Consider the following while working with portlet producers:

- Some out-of-the-box producers are provided with WebCenter Portal, like, WSRP Tools. The `wsrp-tools.ear` EAR files are packaged with WebCenter Portal: for WSRP Tools.

  You can install the `wsrp-tools.ear` files using the `registerOOTBProducers` WLST command. For command syntax and examples, see `registerOOTBProducers` in *WebCenter WLST Command Reference Reference*.
- Before users can add JSR 286 portlets to a page, you must register the owning WSRP producers. See `registerSampleProducers` in *WebCenter WLST Command Reference 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 Extending an Existing Domain in *Installing and Configuring Oracle WebCenter Portal* to install it in the production environment.

- When you create a connection to a portlet producer, the producer is registered with WebCenter Portal 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`. During 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.
- Portlet producer registration is dynamic. New portlet producers and updates to existing producers are immediately available in WebCenter Portal; it is not necessary to restart WebCenter Portal or the managed server.
- To migrate producers from one instance to another, use the migration utilities described in Migrating a WSRP Producer Persistence Store in *Developing for Oracle WebCenter Portal*.
- For information on securing portlet producers, see Securing a WSRP Producer.
- For information about portlet producers created using Oracle PDK-Java, see Managing Oracle PDK-Java Portlet Producers.

Managing WSRP Portlet Producers

System administrators can use WebCenter Portal Administration, Fusion Middleware Control, or the WLST command-line tool to register and manage Web Services for Remote Portlets (WSRP) portlet producers for WebCenter Portal.

- Registering an Oracle PDK-Java Portlet Producer
- Testing Oracle PDK-Java Producer Connections
- Editing Oracle PDK-Java Portlet Producer Registration Details
- Deregistering an Oracle PDK-Java Portlet Producer

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 WebCenter Portal to communicate with the producer and with the portlets through the producer.

WebCenter Portal supports both WSPR 1.0 and WSRP 2.0 producers. The WSRP 2.0 standard provides support for, among other things, 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.

WebCenter Portal provides several tools for registering WSRP portlet producers with deployed applications.Editing Oracle PDK-Java Portlet Producer Registration Details Using Fusion Middleware Control

This section includes the following topics:

- Registering a WSRP Producer Using Fusion Middleware Control

- Registering a WSRP Producer Using WLST
- Registering a WSRP Portlet Producer in WebCenter Portal
- Adding a Grant to the Policy Store for a Mapped User Identity

You can also register portlet producers that have been developed using Oracle PDK-Java. For more information, see Registering an Oracle PDK-Java Portlet Producer.

Registering a WSRP Producer Using Fusion Middleware Control

You can register a WSRP portlet producer using Fusion Middleware Control.

To register a WSRP portlet producer using Fusion Middleware Control:

1. Log in to Fusion Middleware Control and navigate to the home page for the WebCenter Portal application. See Navigating to the Home Page for WebCenter Portal.
2. From the **WebCenter Portal** menu, select **Settings** and then **Service Configuration**.
3. In the Add Portlet Producer Connection section, enter connection details for the WSRP producer.

   For detailed parameter information, see WSRP Producer Connection Parameters.
4. Use the **Security** section to specify the type of security token to use for the identity propagation/assertion.

   For detailed parameter information, see WSRP Producer Security Connection Parameters.

   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 WebCenter Portal. WebCenter Portal supports six types of security token:

   - 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
   - WSS 1.1 SAML Token With Message Protection

   SAML is an abbreviation for Security Assertion Markup Language. For more information about each of these security tokens, see WSRP Producer Security Connection Parameters.

> **Note:**
>
> PeopleSoft WSRP producers support two profiles: Username Token With Password and 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.

5. Use the Keystore section to specify the location of the keystore 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 WSRP Producer Keystore Connection Parameters.

6. Click **OK**.

   The new producer appears in the connection table.

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 WebCenter Portal.

> **Note:**
>
> When you use the WLST command `listWSRPProducers`, you must edit the `$ORACLE_HOME/oracle_common/common/bin/setWlstEnv.sh` and append the following to JVM_ARGS:
>
> `-"Dcom.sun.xml.namespace.QName.useCompatibleSerialVersionUID=1.0"`

For command syntax and examples, see `registerWSRPProducer` in the *WebCenter WLST Command Reference Reference*.

> **See Also:**
>
> `deregisterWSRPProducer`, `listWSRPProducers`, `refreshProducer`, `registerOOTBProducers`, `registerSampleProducers`

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

Registering a WSRP Portlet Producer in WebCenter Portal

You can register a WSRP portlet producer in WebCenter Portal Administration.

To register a WSRP producer in WebCenter Portal:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Tools and Services**.

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

   `http://host:port/webcenter/portal/admin/settings/tools`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click **Portlet Producers**.
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 WSRP Producer Connection Parameters.
5. Use the **Security** section to specify the type of security token to use for the identity propagation/assertion.

   For details, see WSRP Producer Security Connection Parameters.

   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 supports six types of security token.

   - 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
   - WSS 1.1 SAML Token With Message Protection

   SAML is an abbreviation for Security Assertion Markup Language.
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. Anything in the path after the *host*:*port* is ignored. To verify whether the producer is accessible, access the producer's test page in your browser. See Testing WSRP Producer Connections.

7. Click **OK**.

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.

To add a grant to the policy store do one of the following:

> **Note:**
>
> Replace ***MyAppID*** with the name of the client application, including the version number if any.

- Add 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>
```

- Grant the permission by running the `grantPermission` WLST command.

  For example:

```
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')
```

  For command syntax and examples, see `grantPermission` in *Infrastructure Security WLST Command Reference*. For information on how to run WLST commands, see Running Oracle WebLogic Scripting Tool (WLST) Commands.

WSRP Producer Connection Parameters

When you register a WSRP portlet producer, there are several connection parameters that you must set.

Table 17-2 WSRP Portlet Producer Connection Parameters

Field	Description
Connection Name	Enter a unique name to identify this portlet producer registration within WebCenter Portal. The name must be unique across all WebCenter Portal connection types. The name you specify here appears in the resource catalog (under the **Portlets** folder).
Producer Type	Select **WSRP Producer**.

WSDL URL	Enter 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:port/context_root/portlets/wsrp2?WSDL`
	`http://host:port/context_root/portlets/wsrp1?WSDL`
	`http://host:port/context_root/portlets/?WSDL` (WSRP 1.0 for backward compatibility)
	Where:
	• *host* is the server where your producer is deployed.
	• *port* 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 version 2 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:port/context_root/info`

Use Proxy?	Select if WebCenter Portal must use an HTTP proxy when contacting this producer. If selected, enter values for **Proxy Host** and **Proxy Port**. A proxy is required when WebCenter Portal and the remote portlet producer are separated by a firewall and an HTTP proxy is needed to communicate with the producer.

Table 17-2 (Cont.) WSRP Portlet Producer Connection Parameters

Field	Description
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 pages. The default is `30` seconds. Individual portlets may define their own timeout period, which takes precedence over the value expressed here.

WSRP Producer Security Connection Parameters

When you register a WSRP portlet producer, there are some security settings that you can specify.

Table 17-3 WSRP Portlet 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 17-4
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**.

Table 17-3 (Cont.) WSRP Portlet Producer Security Connection Parameters

Field	Description
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.
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, WebCenter Portal 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 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** dropdown list to identify the application. If the application you want is not listed, select **Create New** to define the external application now. Valid for: WSS 1.0 Username With Password only.

Table 17-4 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.
WSS 1.0 Username Token Without Password `oracle/wss10_username_id_propagation_with_msg_protection_client_policy`	This policy provides user name (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 (user name 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 user name (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.

Table 17-4 (Cont.) Token Profiles Options

Token Profile	Description
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.

WSRP Producer Keystore Connection Parameters

When you register a WSRP portlet producer, you can specify the location of the keystore that contains the certificate and private key that is used for signing the SOAP message.

Table 17-5 WSRP Producer Key Store Connection Parameters

Field	Description
Recipient Alias	Specify the keystore alias that is associated with the producer's certificate. This certificate is used to encrypt the message to the producer.
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.

Table 17-5 (Cont.) WSRP Producer Key Store Connection Parameters

Field	Description
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 keystore. 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.

Testing WSRP Producer Connections

You can test a WSRP portlet producer connection to confirm that the producer is up and running.

1. Obtain the producer URL from:

   `http://host:port/context_root/info`

   For a WSRP v2 producer connection, the producer URL format is:

   `http://host:port/context_root/portlets/wsrp2?WSDL`

   For example:

   `http://example.com:7778/MyPortletApp/portlets/wsrp2?WSDL`

   For a WSRP v1 producer connection, the producer URL format is:

   `http://host:port/context_root/portlets/wsrp1?WSDL`

   For example:

   `http://example.com:7778/MyPortletApp/portlets/wsrp1?WSDL`

2. Run the producer URL in a browser window.

Editing WSRP Producer Registration Details

WebCenter Portal provides several tools for editing WSRP portlet producer registration details.

This section includes the following topics:

- About Editing WSRP Producer Registration Details
- Editing WSRP Producer Registration Details Using Fusion Middleware Control

- Editing Producer Registration Details Using WLST
- Editing WSRP Producer Registration Details in WebCenter Portal
- Migrating WSRP Producer Metadata to a New WSDL URL
- Editing the Portlet Client Configuration

For information about how to edit Oracle PDK-Java portlet producer registration details, see Editing Oracle PDK-Java Portlet Producer Registration Details.

About Editing WSRP 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 WDSL URL property.

To retain all the portlet customizations and personalizations that users make while working with WebCenter Portal, 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.

See Backing Up and Restoring Portlet Producer Metadata.

Editing WSRP Producer Registration Details Using Fusion Middleware Control

You can edit WSRP producer registration details using Fusion Middleware Control.

To update connection details for a portlet producer using Fusion Middleware Control:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal. See Navigating to the Home Page for WebCenter Portal.
2. From the **WebCenter Portal** menu, select **Settings** and then **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 WSRP Producer Connection Parameters.
6. Click **OK**.

Editing Producer Registration Details Using WLST

Use the WLST command `setWSRPProducer` to edit WSRP portlet producer connection details.

For command syntax and examples, see `setWSRPProducer` in *WebCenter WLST Command Reference Reference*.

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

Editing WSRP 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 WSRP portlet producer registration details in WebCenter Portal:

1. Open WebCenter Portal Administration.

   For more information, see Accessing the Settings Pages in WebCenter Portal Administration.

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

   For details, see WSRP Producer Connection Parameters and WSRP Producer Security Connection Parameters.

   You cannot edit the **Producer Name** or **Producer Type**.

   > **Note:**
   >
   > While it is possible to edit the value of the **WSDL URL**, 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 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. Anything in the path after the *host*:*port* is ignored. To verify whether the producer is accessible, access the producer's test page in your browser. See Testing WSRP Producer Connections.

7. Click **OK**.

Migrating WSRP Producer Metadata to a New WSDL URL

If a producer moves to a different location, then to retain all the portlet customizations and personalizations that users have made while working with WebCenter Portal, you must also migrate the existing producer metadata to the new location. Any existing connections to the producer must be reconfigured to point 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 `exportPortletClientMetadata` in *WebCenter WLST Command Reference Reference*.

2. Change the producer's WSDL URL, using the WLST command `setWSRPProducer`.

   For command syntax and examples, see `setWSRPProducer` in *WebCenter WLST Command Reference Reference*.

3. Import the producer metadata, using the WLST command `importPortletClientMetadata`.

   For command syntax and examples, see `importPortletClientMetadata` in *WebCenter WLST Command Reference Reference*.

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.

The following example shows the `adf-portlet-config` element of the `adf-config.xml` file.

```
<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.WebPortletTechnologyConf
ig</value>
<value>oracle.portlet.client.containerimpl.wsrp.WSRPPortletTechnologyCo
nfig</value>
  </portletTechnologies>
  <defaultTimeout>20</defaultTimeout>
  <minimumTimeout>1</minimumTimeout>
  <maximumTimeout>300</maximumTimeout>
  <resourceProxyPath>/portletresource</resourceProxyPath>
  <cacheSettings enabled="true">
    <serviceConfigFile>myPortletCoherenceConfig.xml</serviceConfigFile>
```

```
  </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 `setPortletClientConfig` section in *WebCenter WLST Command Reference Reference*.

After using this WLST command, you must restart the Managed Server on which the WebCenter Portal application is deployed. For details, see Starting and Stopping Managed Servers for WebCenter Portal Application Deployments.

> **See Also:**
>
> `listPortletClientConfig`, `getPortletClientConfig`
>
> For information on how to run WLST commands, see Running Oracle WebLogic Scripting Tool (WLST) Commands.

Deregistering WSRP Portlet Producers

WebCenter Portal provides several tools for deregistering WSRP portlet producers.

This section includes the following topics:

- About Deregistering Portlet Producers
- Deregistering a WSRP Portlet Producer Using Fusion Middleware Control
- Deregistering a WSRP Portlet Producer Using WLST
- Deregistering a WSRP Portlet Producer in WebCenter Portal

For information about how to deregister Oracle PDK-Java portlet producers, see Deregistering an Oracle PDK-Java Portlet Producer.

About Deregistering Portlet Producers

You can deregister a WSRP portlet producer at any time.

Before deregistering a producer, consider the impact to WebCenter Portal 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 Viewing Performance Metrics Using Fusion Middleware Control.

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).

Portlet instances are not removed from WebCenter Portal pages. In place of the portlet, users see a `Portlet unavailable` message.

> **Note:**
>
> Consider also deleting the external application associated with this portlet producer *if* the application's sole purpose was to support this producer. See Deleting External Application Connections.

Deregistering a WSRP Portlet Producer Using Fusion Middleware Control

You can deregister a WSRP portlet producer using Fusion Middleware Control.

To deregister a portlet producer:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal.

   See Navigating to the Home Page for WebCenter Portal.
2. From the **WebCenter Portal** menu, select **Settings** and then **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 deregister, and click **Delete**.

The connection details are removed. Portlets associated with this producer are no longer accessible within WebCenter Portal.

Deregistering a WSRP Portlet Producer Using WLST

Use WLST commands to deregister a WSRP portlet producer.

Use the WLST command `deregisterWSRPProducer` to deregister a WSRP portlet producer connections.

For command syntax and examples, see `deregisterWSRPProducer` in *WebCenter WLST Command Reference Reference*.

Use the following WLST commands to deregister the out-of-the-box or sample producers provided with WebCenter Portal:

- **Out-of-the-box producers** - `deregisterOOTBProducers`

  For command syntax and examples, see `deregisterOOTBProducers` in *WebCenter WLST Command Reference Reference*.
- **Sample producers** - `deregisterSampleProducers`

  For command syntax and examples, see `deregisterSampleProducers` in *WebCenter WLST Command Reference Reference*.

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

Deregistering a WSRP Portlet Producer in WebCenter Portal

If you no longer want to use a particular producer in WebCenter Portal, you can deregister the producer.

To deregister a WSRP portlet producer in WebCenter Portal:

1. Open WebCenter Portal Administration.

   For more information, see Accessing the Settings Pages in WebCenter Portal Administration.

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/settings/tools`

3. Select the portlet producer that you want to deregister.
4. From the menu bar, click **Deregister**.
5. In the Delete Confirmation dialog, click **Deregister** to complete the deregistration process.

Managing Oracle PDK-Java Portlet Producers

System administrators can use Fusion Middleware Control or the WLST command-line tool to register and manage Oracle PDK-Java portlet producers for WebCenter Portal.

This section includes the following topics:

- Registering an Oracle PDK-Java Portlet Producer
- Testing Oracle PDK-Java Producer Connections
- Editing Oracle PDK-Java Portlet Producer Registration Details
- Deregistering an Oracle PDK-Java Portlet Producer

Registering an Oracle PDK-Java Portlet Producer

You can register an Oracle PDK-Java portlet producer using Fusion Middleware Control, WLST commands, or WebCenter Portal Administration.

This section includes the following topics:

- Registering an Oracle PDK-Java Portlet Producer Using Fusion Middleware Control
- Registering an Oracle PDK-Java Portlet Producer Using WLST
- Registering an Oracle PDK-Java Portlet Producer in WebCenter Portal

Registering an Oracle PDK-Java Portlet Producer Using Fusion Middleware Control

To register an Oracle PDK-Java portlet producer using Fusion Middleware Control:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal.

See Navigating to the Home Page for WebCenter Portal.

2. From the **WebCenter Portal** menu, select **Settings** and then **Service Configuration**.
3. In the **Add Portlet Producer Connection** section, enter connection details for the Oracle PDK-Java portlet producer.

   For detailed parameter information, see Oracle PDK-Java Portlet Producer Connection Parameters.
4. Click **OK**.

   The new producer appears in the connection table.

Registering an Oracle PDK-Java Portlet Producer Using WLST

Use the WLST command `registerPDKJavaProducer` to create a connection to an Oracle PDK-Java portlet producer and register the producer with WebCenter Portal.

For command syntax and examples, see `registerPDKJavaProducer` in *WebCenter WLST Command Reference Reference*.

> **See Also:**
>
> `deregisterPDKJavaProducer`, `listPDKJavaProducers`, `refreshProducer`

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

Registering an Oracle PDK-Java Portlet Producer in WebCenter Portal

To register an Oracle PDK-Java portlet producer in WebCenter Portal:

1. Open WebCenter Portal Administration.
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/settings/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 Oracle PDK-Java Portlet Producer Connection Parameters.
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. Anything in the path after the *host*:*port* is ignored. To verify whether the producer is accessible, access the producer's test page in your browser. See Testing Oracle PDK-Java Producer Connections.

6. Click **OK**.

Testing Oracle PDK-Java Producer Connections

To verify an Oracle PDK-Java portlet producer connection, run the producer URL in a browser window.

Use the following format:

```
http://host:port/context-root/providers/producer_name
```

For example:

```
http://domain.example.com:7778/axyz/providers/sample
```

Editing Oracle PDK-Java Portlet Producer Registration Details

WebCenter Portal provides several tools for editing Oracle PDK-Java portlet producer registration details.

This section includes the following topics:

- Editing Oracle PDK-Java Portlet Producer Registration Details Using Fusion Middleware Control
- Editing Oracle PDK-Java Portlet Producer Registration Details Using WLST
- Editing Oracle PDK-Java Portlet Producer Registration Details in WebCenter Portal

Editing Oracle PDK-Java Portlet Producer Registration Details Using Fusion Middleware Control

1. Log in to Fusion Middleware Control and navigate to the home page for the WebCenter Portal application. For more information, see:

   For more information, see Navigating to the Home Page for WebCenter Portal.
2. From the WebCenter Portal menu, select **Settings** and then **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 Oracle PDK-Java Portlet Producer Connection Parameters.

6. Click **OK**.

Editing Oracle PDK-Java Portlet Producer Registration Details Using WLST

Use the WLST command `setPDKJavaProducer` to edit Oracle PDK-Java portlet producer connection details.

For command syntax and examples, see `setPDKJavaProducer` in *WebCenter WLST Command Reference Reference*.

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

Editing Oracle PDK-Java Portlet 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 PDK-Java portlet producer registration details in WebCenter Portal:

1. Open WebCenter Portal Administration.

   For more information, see Accessing the Settings Pages in WebCenter Portal Administration.

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.

   For details, see Oracle PDK-Java Portlet Producer Connection Parameters.

   You cannot edit the **Producer Name** or **Producer Type**.

   > **Note:**
   >
   > While it is possible to edit the value of the **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 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. Anything in the path after the *host*:*port* is ignored. To verify whether the producer is accessible, access the producer's test page in your browser. See Testing Oracle PDK-Java Producer Connections.

Deregistering an Oracle PDK-Java Portlet Producer

WebCenter Portal provides several tools for deregistering Oracle PDK-Java portlet producers.

This section includes the following topics:

- About Deregistering an Oracle PDK-Java Portlet Producer
- Deregistering an Oracle PDK-Java Portlet Producer Using Fusion Middleware Control
- Deregistering an Oracle PDK-Java Portlet Producer Using WLST
- Deregistering an Oracle PDK-Java Portlet Producer in WebCenter Portal

About Deregistering an Oracle PDK-Java Portlet Producer

You can deregister an Oracle PDK-Java Portlet Producer portlet producer at any time.

Before deregistering a producer, consider the impact to WebCenter Portal 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 Viewing Performance Metrics Using Fusion Middleware Control.

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).

Portlet instances are not removed from WebCenter Portal pages. In place of the portlet, users see a `Portlet unavailable` message.

> **Note:**
>
> Consider also deleting the external application associated with this portlet producer *if* the application's sole purpose was to support this producer. See Deleting External Application Connections.

Deregistering an Oracle PDK-Java Portlet Producer Using Fusion Middleware Control

To deregister an Oracle PDK-Java portlet producer using Fusion Middleware Control:

1. Log in to Fusion Middleware Control and navigate to the home page for WebCenter Portal.

   See Navigating to the Home Page for WebCenter Portal

2. From the **WebCenter Portal** menu, select **Settings** and then **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 deregister, and click **Delete**.

   The connection details are removed. Portlets associated with this producer are no longer accessible within WebCenter Portal.

Deregistering an Oracle PDK-Java Portlet Producer Using WLST

Use the WLST command `deregisterPDKProducer` to deregister an Oracle PDK-Java portlet producer.

For command syntax and examples, see `deregisterPDKJavaProducer` in *WebCenter WLST Command Reference Reference*.

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

Deregistering an Oracle PDK-Java Portlet Producer in WebCenter Portal

To deregister an Oracle PDK-Java portlet producer in WebCenter Portal

1. Open WebCenter Portal Administration.

   For more information, see Accessing the Settings Pages in WebCenter Portal Administration.

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/settings/tools`

3. Select the portlet producer that you want to deregister.
4. On the menu bar, click **Deregister**.
5. In the Delete Confirmation dialog, click **Deregister** to complete the deregistration process.

Oracle PDK-Java Portlet Producer Connection Parameters

When you register an Oracle PDK-Java portlet producer, there are several connection parameters that you must set.

Table 17-6 Oracle PDK-Java Portlet Producer Connection Parameters

Field	Description
Connection Name	Enter a unique name that identifies this portlet producer registration within WebCenter Portal. The name must be unique across all WebCenter Portal connection types. The name you specify here appears in the resource catalog (under the **Portlets** folder).
Producer Type	Select **Oracle PDK-Java Producer**.

URL End Point	Enter the Oracle PDK-Java producer's URL using the following syntax:
	`http://host:port/context_root/providers`
	Where:
	• *host* is the server where the producer is deployed
	• *port* 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 WebCenter Portal must use an HTTP proxy when contacting this producer. If selected, enter values for **Proxy Host** and **Proxy Port**. A proxy is required if WebCenter Portal and the remote portlet producer are separated by a firewall and an HTTP proxy is needed for communication with the producer.

Table 17-6 (Cont.) Oracle PDK-Java Portlet Producer Connection Parameters

Field	Description
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 Registering External Applications.
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 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 WebCenter Portal, a call is made to the producer. During the call, the consumer (WebCenter Portal 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. The Shared Key is also known as the HMAC key.

Deploying Portlet Producer Applications

After developing a Portlet Producer application in JDeveloper, you can deploy that application to any Oracle WebLogic Managed Server instance that is configured to supportWebCenter 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 Oracle WebCenter Portal Administration Tools.

You can also deploy a Portlet Producer application from within JDeveloper.

This section includes the following topics:

- Preparing Portlet Producer Applications for Deployment
- Deploying a Portlet Producer Application Using Fusion Middleware Control
- Deploying a Portlet Producer Application Using Oracle WebLogic Server Administration Console
- Deploying a Portlet Producer Application Using WLST
- Deploying a Portlet Producer Application Using Oracle JDeveloper

For more information about deploying applications, see Deploying Applications in *Administering Oracle Fusion Middleware*.

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.

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.

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.

Deploying a Portlet Producer Application Using Fusion Middleware Control

You can deploy a Portlet Producer using Fusion Middleware Control.

When deploying an application using Fusion Middleware Control you must know the location of the application archive, and whether a deployment plan exists for the application.

To deploy a Portlet Producer application using Fusion Middleware Control:

1. Log in to Fusion Middleware Control.
2. In the Target Navigation pane, expand **WebLogic Domain** and click the domain in which your target Managed Server was created.
3. From the WebLogic Domain menu, select **Deployments**.

4. Choose **Deployment > Deploy**.

   The Select Archive page displays.

5. 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.

6. 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.

7. Click **Next**.

   The Select Target page displays.

8. Select the target server(s) to deploy the application and click **Next**.

   The Application Attributes page displays.

9. Click Next.

   The Deployment Settings page displays.

10. Click the **edit** icon for Configure ADF Connections to check connection settings associated with the application.

    The Configure ADF Connections page displays.

11. 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 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.

> **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 or WLST commands to modify the producer URL endpoint, and then redeploy the application.

12. If required, specify additional deployment options such as the Web modules to include in your application or security migration settings.
13. In the Deployment Plan section, click **Edit Deployment Plan** to optionally edit the currently selected Deployment Plan.
14. In the Deployment Plan section, click **Save Deployment Plan** to optionally save the currently selected Deployment Plan for reuse when you redeploy the application.
15. To start the deployment process, click **Deploy**.

    Fusion Middleware Control displays processing messages.
16. Click **Close** in the Deployment Succeeded page.

    The portlet producer application (and its deployment plan) is now deployed on the WebLogic Managed Server instance.
17. 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.

    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.

Deploying a Portlet Producer Application Using Oracle WebLogic Server Administration Console

You can use the WebLogic Server Administration Console to deploy 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 Portlet Producer application, the MDS connection in the EAR file must be configured to the target deployment repository. Follow steps 1-5 in Deploying a Portlet Producer Application Using WLST then follow the steps below to deploy a Portlet Producer application using the WLS Administration Console.

> **Note:**
>
> Oracle does not recommend deploying Portlet Producer applications to any of the preconfigured Managed Servers created during the installation, or to the Administration Server. Create a new Managed Server instance before deploying, or optionally deploy to the `WC_Portlet` server.

To deploy a Portlet Producer application using the Web Logic Server Administration Console:

1. Log in to the Web Logic Server Administration Console.
2. In the Domain Structure pane, click **Deployments**.

   The Summary of Deployments page displays.
3. Click **Install**.

   The Install Application Assistant page displays.
4. Using the Install Application Assistant **Path** field, locate the EAR file that corresponds to the Portlet Producer application you want to install. Select the EAR file and click **Next**.

   Page 2 of the Install Application Assistant page displays.
5. Select **Install this deployment as an application** and click **Next**.

   Page 3 of the Install Application Assistant displays.
6. Select the deployment target to which to deploy the 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 or WLST commands to modify the producer URL endpoint, and then redeploy the application.

Deploying a Portlet Producer Application Using WLST

You can deploy a Portlet Producer application using the WLST command line.

To deploy a Portlet Producer 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 Portlet Producer Application using WLST:

1. Start the WLST shell.

   For information on starting the WLST shell, see 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, welcome1).
- *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 getMDSArchiveConfig in *WLST Command Reference for Infrastructure Components*.

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 Portlet Producer application using the WLST deploy command.

```
deploy(app_name, path, [targets] [stageMode], [planPath], [options])
```

Where:

- appName is the name of the Portlet Producer application to be deployed (for example, myPortlets).

- `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, `AppServer`). 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 WLST `deploy` in *WLST Command Reference for WebLogic Server*.

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 or WLST commands to modify the producer URL endpoint, and then redeploy the application.

Deploying a Portlet Producer Application Using Oracle JDeveloper

You can deploy a Portlet Producer application 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.

18

Managing External Applications

Register and manage external applications for WebCenter Portal deployments.
An external application is any application that implements its own authentication process. Specifically, it is an application that does not take part in the single sign-on process for WebCenter Portal.

Application administrators can register and manage external applications using Fusion Middleware Control or the WLST command-line tool, or at runtime through built-in administration pages or using external application task flows.

All external application changes that you make for WebCenter Portal post deployment, are stored in the MDS repository as customizations.

> **Note:**
>
> External application configuration is dynamic. Configuration changes are immediately reflected in WebCenter Portal; it is not necessary to restart the application or the managed server.

> **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 granted through WebCenter Portal Administration.
>
> For more information about roles and permissions, see Understanding Administrative Operations, Roles, and Tools.

Topics:

- About External Applications
- Registering External Applications
- Modifying External Application Connection Details
- Deleting External Application Connections
- Managing External Applications at Runtime

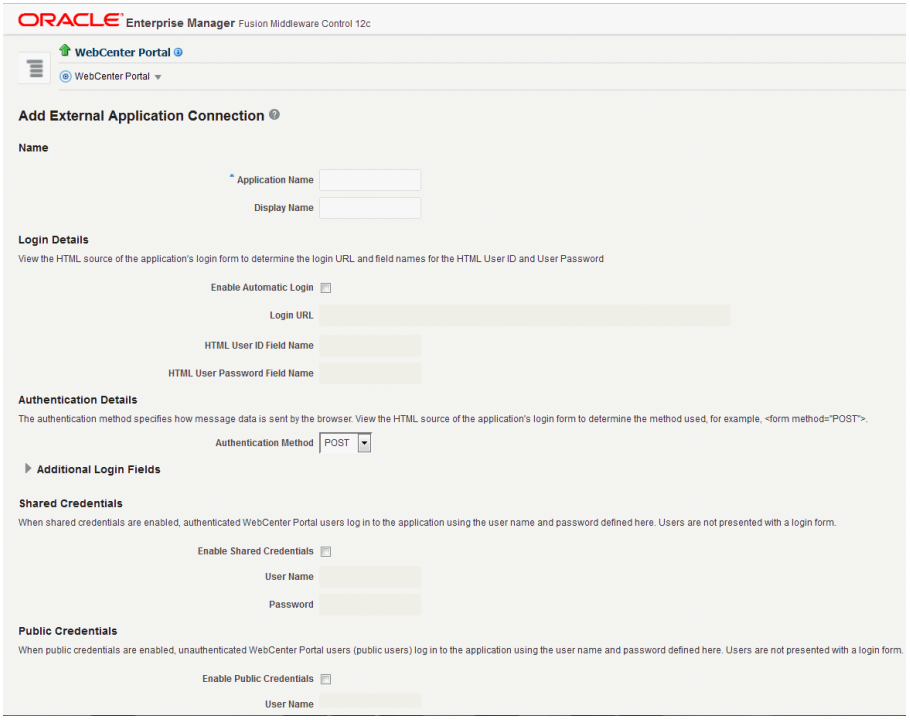
About External Applications

If WebCenter Portal 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.

Figure 18-1 Add External Application Connection



Registering External Applications

You can register external applications for WebCenter Portal 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.

This section includes the steps for:

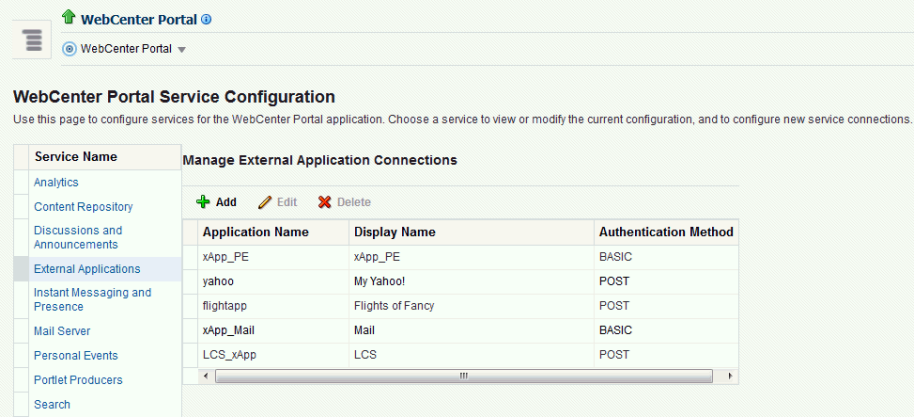
- Registering External Applications Using Fusion Middleware Control
- Registering External Applications Using WLST

For information about registering external applications through WebCenter Portal Administration, see Registering External Applications at Runtime.

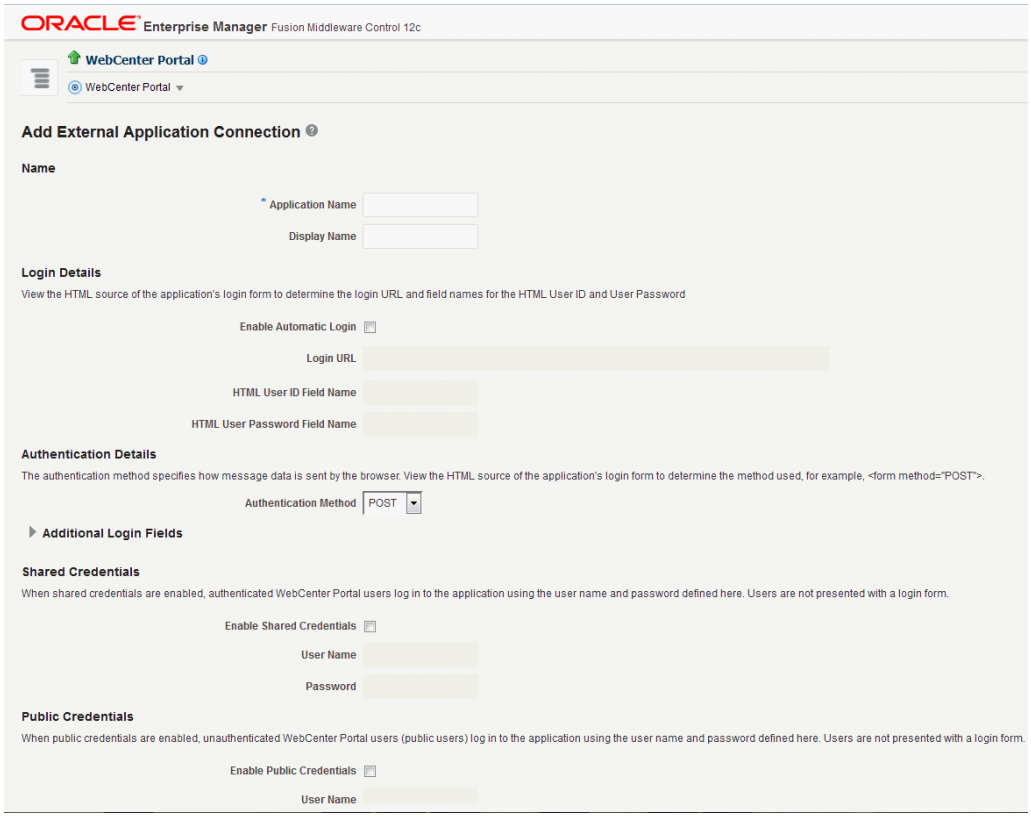
Registering External Applications Using Fusion Middleware Control

To register an external application:

1. Log in to Fusion Middleware Control and navigate to the home page for your WebCenter Portal instance.
2. From the **WebCenter Portal** menu, select **Settings** , then **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**.



5. Enter a unique name for the external application and a display name that application users working with this external application sees.

Table 18-1 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 the login details for the external application.

Table 18-2 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.
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**).

Table 18-2 (Cont.) External Application Connection - Login Details

Field	Description
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.

Table 18-3 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.

Table 18-4 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. Optional: Specify shared and public user credentials, if required.

Table 18-5 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 WebCenter Portal 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*.
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 WebCenter Portal's public pages are logged in using the user name 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.

Registering External Applications Using WLST

Use the WLST command `createExtAppConnection` to create an external application connection. For command syntax and examples, see `createExtAppConnection` in *WebCenter WLST Command Reference Reference*.

Use the WLST command `addExtAppCredential` to add shared or public credentials for an existing external application connection. For more information, see `addExtAppCredential` in *WebCenter WLST Command Reference Reference*.

Use the WLST command `addExtAppField` to define additional login criteria for an existing external application connection. For more information, see `addExtAppField` in *WebCenter WLST Command Reference Reference*.

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

Modifying External Application Connection Details

This section shows you how to modify the external application connection details by:

- Modifying External Application Connection Using Fusion Middleware Control
- Modifying External Application Connection Using WLST

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 application.
2. From the **WebCenter Portal** menu, select **Settings** > **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 the connection details, as required. For detailed parameter information, see Table 18-2. Note that you cannot edit the name of the external application.
6. Click **OK** to save your changes.

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 *WebCenter WLST Command Reference 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 `removeExtAppField`.

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

For information about modifying external applications in WebCenter Portal, see *Editing External Application Connection Details* in *Using Portals in Oracle WebCenter Portal*.

Deleting External Application Connections

Take care when deleting an external application connection as users in WebCenter Portal 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 topics:

- Deleting External Application Connections Using Fusion Middleware Control
- Deleting External Application Connections Using WLST

Deleting External Application Connections Using Fusion Middleware Control

To delete an external application connection:

1. Log in to Fusion Middleware Control and navigate to the home page for your WebCenter Portal application:
2. From the **WebCenter Portal** menu, select **Settings** > **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**.

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 *WebCenter WLST Command Reference 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 Running Oracle WebLogic Scripting Tool (WLST) Commands.

Managing External Applications at Runtime

An external application is any application that implements its own authentication process. Specifically, it is an application that does not take part in the WebCenter Portal application's single sign-on process. If your WebCenter Portal application interacts with an application that handles its own authentication, you can register an external application to allow for credential provisioning.

By default, users with the `Administrator` role have the `AppConnectionManager` role; and therefore, can configure and manage external applications through the WebCenter Portal Administration Console at runtime. For more information about `AppConnectionManager` role, see Default Application Roles.

This section includes the following topics:

- Registering External Applications at Runtime
- Editing and Deleting External Applications at Runtime

Registering External Applications at Runtime

To register an external application at runtime:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Tools and Services**.

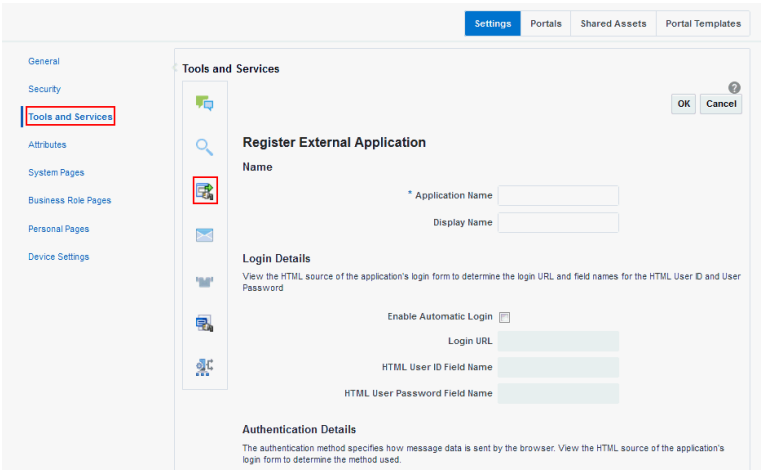
   You can also enter the following URL in your browser to navigate directly to the **Tools and Services** pages:

   `http://host:port/webcenter/portal/admin/settings/tools`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click **External Applications**.
3. Click **Register**.

Figure 18-2 WebCenter Portal Administration Console - External Applications



4. Enter connection details for the external application.

   If you need help with one or more fields, refer to:

   - Table 18-1
   - Table 18-2
   - Table 18-3
   - Table 18-4
   - Table 18-5

5. Click **Test** to verify your connection details.
6. Click **OK** to register the application.

Editing and Deleting External Applications at Runtime

To modify or delete external applications at runtime:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Tools and Services**.

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

   `http://host:port/webcenter/portal/admin/settings/tools`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click **External Applications**.
3. Select the external application to edit or delete, then click one of the following:
   - Click **Edit** to update connection details.
   - Click **Deregister** to remove the external application.

Take care when deleting an external application connection as users will no longer have access to that application, and any services dependent on the external application may not function correctly.

19

Managing REST Services

Use REST services to access many WebCenter Portal tools and services, such as lists, people connections, and search.

> **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 through WebCenter Portal Administration.
>
> For more information about roles and permissions, see Understanding Administrative Operations, Roles, and Tools.

Topics:

- About REST Services
- Performing Required Manual Configurations to Enable REST
- Understanding Security Tokens
- Changing the REST Root Name
- Using Compression
- Handling Authentication

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.

WebCenter Portal provides a REST interface to many of its tools and services, like lists, people connections, and search. For a complete list of the services that support REST and a more complete introduction to REST and Oracle WebCenter Portal REST APIs, see Using the WebCenter Portal REST APIs in *Developing for Oracle WebCenter Portal*.

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 *Developing for Oracle WebCenter Portal*.

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.

- Configuring an Identity Asserter
- Configuring the WebLogic Server Credential Store

Configuring an Identity Asserter

You must configure an identity asserter before using the REST APIs. For detailed instructions, see Configuring the REST Service Identity Asserter.

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")
```

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 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 *Developing for Oracle WebCenter Portal*.

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 URL Rewriting. For more information, see URL Rewriting and Proxy Server Capabilities in *Administering Oracle HTTP Server*. For example, after URL Rewriting, the following REST API URLs point to the same server:

- `http://myhost:8888/rest/api/resourceIndex`
- `http://myhost:8888/pathname/rest/api/resourceIndex`

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 Oracle HTTP Server (OHS) or running Apache, you can add the HTTP request header `Accept-Encoding: gzip, deflate` to use the compression in Rest API response.

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 *Oracle Fusion Middleware Administrators Guide for Oracle HTTP Server*.

For more information on Oracle Web Cache, see Compression and Caching and Compressing Content in *Oracle Fusion Middleware Administrators Guide for Oracle HTTP Server*.

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 Configuring the REST Service Identity Asserter.

For more information, see Configuring Authentication Providers in *Administering Security for Oracle WebLogic Server*.

Part IV
Monitoring

This part of *Administering Oracle WebCenter Portal* provides information about monitoring Oracle WebCenter Portal using Oracle Enterprise Manager Fusion Middleware Console.

- Monitoring WebCenter Portal Performance
- Managing WebCenter Portal Logs
- Managing WebCenter Portal Audit Logs

20

Monitoring WebCenter Portal Performance

Monitor a range of performance metrics for WebCenter Portal through Fusion Middleware Control, and troubleshoot issues by analyzing information that is recorded in diagnostic log files.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

> **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 Understanding Administrative Operations, Roles, and Tools.

Topics:

- Understanding Oracle WebCenter Portal Performance Metrics
- Viewing Performance Metrics Using Fusion Middleware Control
- Customizing Key Performance Metric Thresholds and Collection
- Diagnosing and Resolving Performance Issues with Oracle WebCenter Portal
- Tuning Oracle WebCenter Portal Performance
- Monitoring Performance Using WebCenter Portal Performance Pack
- Improving Data Caching Performance

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, as well as the application as a whole. To access Oracle WebCenter Portal metrics through Fusion Middleware Control, see 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 Oracle 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:

- Understanding Oracle WebCenter Portal Metric Collection
- Understanding the Key Performance Metrics
- Using Key Performance Metric Data to Analyze and Diagnose System Health
- Understanding Some Common Performance Issues and Actions
- Understanding Page Request Metrics
- Understanding Portlet Producer Metrics
- Understanding WebLogic Server Metrics
- Understanding Security Metrics
- Understanding Page Response and Load Metrics
- Understanding Portal Metrics
- Understanding Tool and Service Metrics

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. 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

Metric Collection: Since Startup

At any given time, real-time metrics are available for the duration for which the WebLogic Server hosting WebCenter Portal 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.

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.

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 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 Understanding the Key Performance Metrics.

Understanding the Key Performance Metrics

Diagnosing the availability and performance of WebCenter Portal 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 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 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 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.

Oracle WebCenter Portal allows you to manage warning thresholds for the key performance metrics described in Table 20-1:

Table 20-1 Key Performance Metric Collection

Component	Key Performance Metric	Metric Sampling
WebCenter Portal	Active Sessions	1 sample every X minutes
WebCenter Portal - **Pages**	Page Response Time	Per Request
WebCenter Portal - **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 and portlets, 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 Configuring the Frequency of WebLogic Server Health Checks.

The total number of samples that Oracle WebCenter Portal collects is configurable too, as described in 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 performance. You can view all key performance metric data from your application's home page in Fusion Middleware Control.

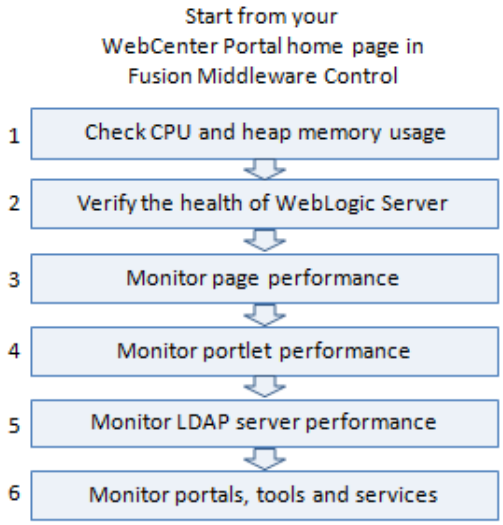
Using Key Performance Metric Data to Analyze and Diagnose System Health

If you monitor WebCenter Portal 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 20-1 presents high-level steps for monitoring the out-of-the-box application *WebCenter Portal*.

Figure 20-1 Analyzing System Health for WebCenter Portal - Main Steps



Note:

- Steps 4 applies only if your application utilizes the portlets feature.
- Bar charts appear grey if a feature is not used.
- Line charts require at least 3 data points before they start to show data.

Table 20-2 Analyzing System Health - Step by Step

Step	Description
Navigate to the home page for WebCenter Portal	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. See Navigating to the Home Page for WebCenter Portal.

Table 20-2 (Cont.) Analyzing System Health - Step by Step

Step	Description
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: Understanding WebLogic Server Metrics • **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 Troubleshooting Slow Requests Using JFR Recordings. **Next Step:** If the charts indicate that CPU and memory usages are normal, verify the health of the WebLogic Server.

Table 20-2 (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 Understanding WebLogic Server Metrics and 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_Portal] [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 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 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` **Next Step**: If the charts indicate that WebLogic Server is performing within thresholds, verify the health of your WebCenter Portal application.

Table 20-2 (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 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_Portal] [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 Viewing and Configuring Log Information. Identify individual pages that are not performing. For details, see 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 How to Troubleshoot Slow Page Requests.

Table 20-2 (Cont.) Analyzing System Health - Step by Step

Step	Description
4. 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 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_Portal] [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 MyPortlets 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_Portlet`), 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. **Next Step:** If the charts indicate that portlet requests are performing within thresholds, verify the performance of your LDAP server.

Table 20-2 (Cont.) Analyzing System Health - Step by Step

Step	Description
5. 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 Understanding Security Metrics. 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>` **Next Step:** If your LDAP server is performing within thresholds, investigate other areas.
6. Monitor individual tools and services	Look at the **WebCenter Portal Services** section at the bottom of the home page. For details, see Understanding Tool and Service Metrics. Quickly see if a particular tool or service is "Down" or "Unknown". Refer to 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.

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 Troubleshooting 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 Administering Tools and Services .

  When you reconfigure connections to tools and services you must always restart the managed server on which the WebCenter Portal 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 Customizing Key Performance Metric Thresholds and Collection.

Understanding Page Request Metrics

You can monitor the availability and performance of page requests for WebCenter Portal through Fusion Middleware Control. You can monitor recent page data and historical (overall) page data.

This section includes the following information:

- Understanding Full Page and Partial Page Metrics
- Recent Page Metrics
- Overall Page Metrics

> **Note:**
>
> The *page request* metrics discussed in this section are different from the *page operation* metrics discussed in 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).

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 Rerendering Partial Page Content in *Developing Web User Interfaces with Oracle ADF Faces*.

Recent Page Metrics

Recent page availability and performance metrics are summarized on the home page for WebCenter Portal (Figure 20-2 and Table 20-3). The page availability/performance charts show at a glance if page requests are slower than expected or failing.

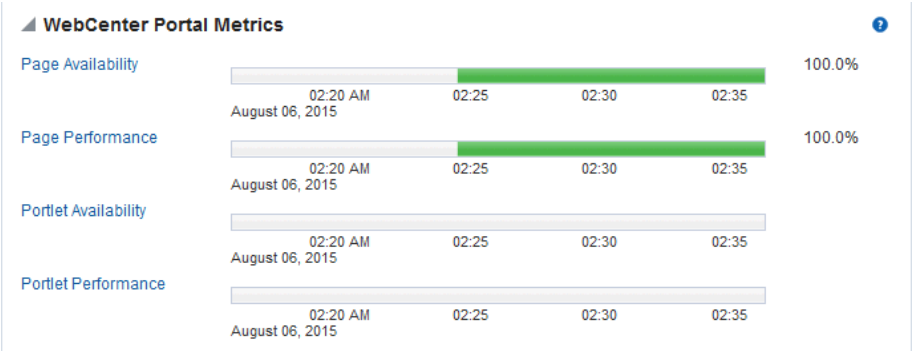
> **Note:**
>
> To access the home page, see Navigating to the Home Page for WebCenter Portal.

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/portlet request time and ends with the current time. See 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 20-2 Recent Page Summary on the WebCenter Portal Home Page

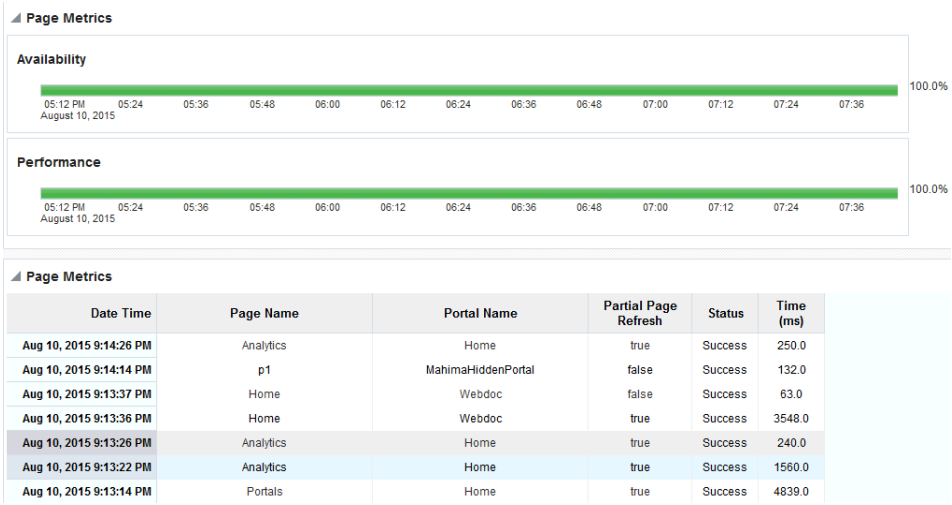


If the chart indicates issues or incidents, click the **Page Availability** or **Page Performance** link to navigate to more detailed information to diagnose the issue further (see Figure 20-3 and Table 20-3).

Use the information on the Recent Page Metrics page (Figure 20-3) 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 Customizing Key Performance Metric Thresholds and Collection.

Figure 20-3 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 20-2.

Table 20-3 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 20-3 (Cont.) Recent Page Request Metrics

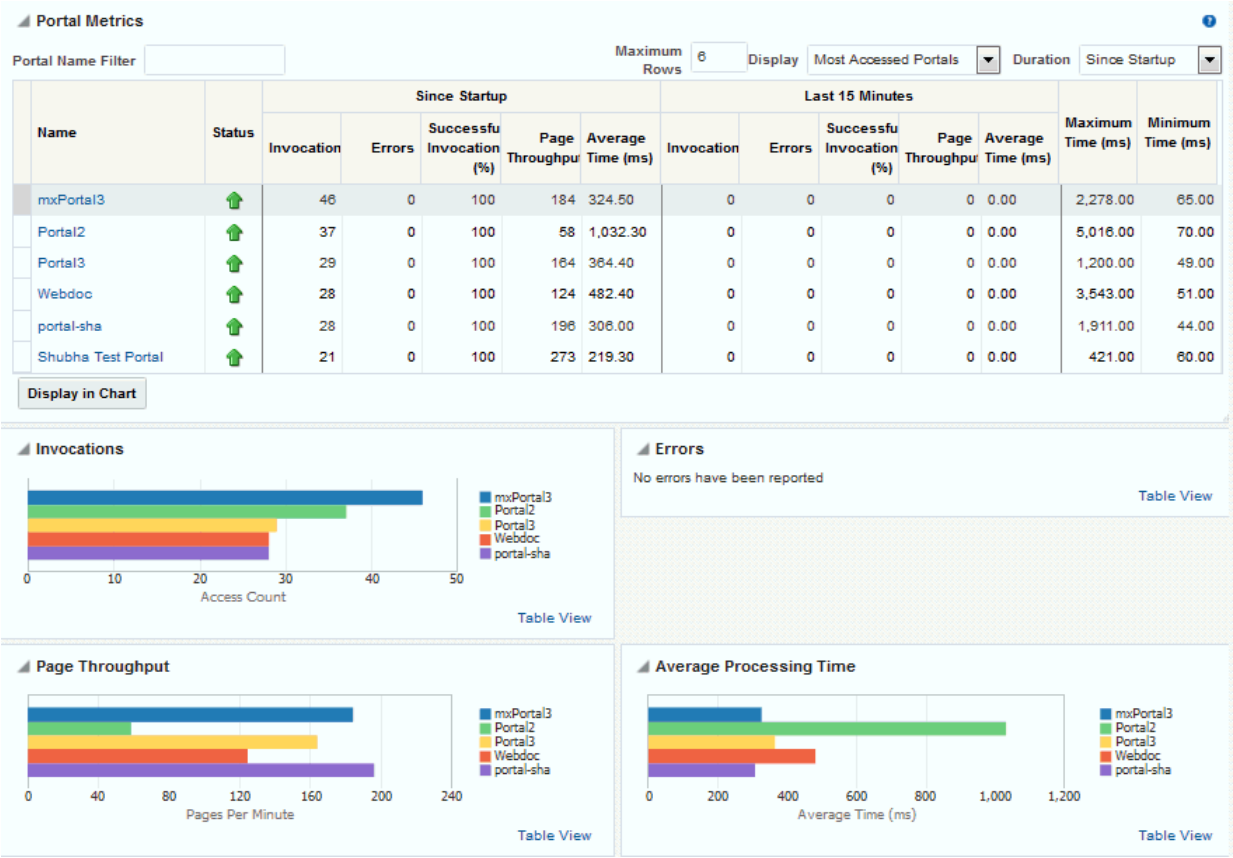
Metric	Description
Performance	Indicates page performance over the last 'N' page requests: • **Green** - Indicates acceptable page response times, that is, the time taken to respond is less than a predefined threshold. • **Red** - Indicates page responses exceeding the limit set. For example, if your 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.
Portal Name	Name of the portal 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".

Overall Page Metrics

Historical performance metrics associated with page activity are also available as shown in Figure 20-4 and described in Table 20-4. 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 Viewing Performance Metrics Using Fusion Middleware Control.

Figure 20-4 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:**
>
> Metrics for pages in the Home portal are not included.

Table 20-4 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. • **Portal 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 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 accessed 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.
Portal Name	Names of portals 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 - Last 15 Minutes
Average Time (ms)	Average time (in ms) to display the page (full or partial): - Since Startup - Last 15 Minutes
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 - Last 15 Minutes If Successful Invocations (%) is below 100%, check the diagnostic logs to establish why page requests are failing. See Viewing and Configuring Log Information.
Pages per Minute	Number of times the page is accessed per minute, also referred to as page throughput: - Since Startup - Last 15 Minutes

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** - Graph showing the number of errors. Use this graph to compare error rates.
- **Average Processing Time** - Graph showing the average page response time (in milliseconds). Use this graph to identify pages with the best (or worst) 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**.

Understanding Portlet Producer Metrics

You can monitor the availability and performance of all the portlets and portlet producers used by WebCenter Portal through Fusion Middleware Control. You can monitor recent and historical (overall) portlet data. The following topics describe the metrics that are available:

- Recent Portlet Metrics
- Overall Portlet Producer Metrics
- Overall Portlet Metrics

Recent Portlet Metrics

Recent portlet availability and performance metrics are summarized on the home page for WebCenter Portal (Figure 20-5 and Table 20-5). The portlet availability/ performance charts show at a glance if portlet requests are slower than expected or failing.

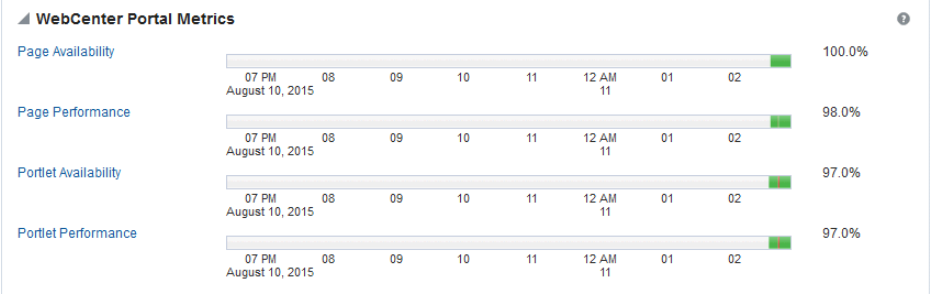
> **Note:**
>
> To access the home page, see Navigating to the Home Page for WebCenter Portal.

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/portlet request time and ends with the current time. See 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 20-5.

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 Recent Page Metrics and the same applies to the portlet charts.

Figure 20-5 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 20-6 and Table 20-5).

Figure 20-6 Recent Portlet Metrics

Portlet Metrics

Date Time	Portlet Name	Producer Name	Producer Type	Status Code	Status	Time (ms)
Aug 11, 2015 2:50:59 AM	Parameter Form	wc-WSRPTools	WSRP	200	Success	43.0
Aug 11, 2015 2:50:59 AM	OmniPortlet	wc-OmniPortlet	Web	200	Success	11.0
Aug 11, 2015 2:50:26 AM	OmniPortlet	wc-OmniPortlet	Web	200	Success	17.0
Aug 11, 2015 2:50:26 AM	Parameter Form	wc-WSRPTools	WSRP	200	Success	32.0
Aug 11, 2015 2:50:23 AM	OmniPortlet	wc-OmniPortlet	Web	200	Success	56.0
Aug 11, 2015 2:50:23 AM	Parameter Form	wc-WSRPTools	WSRP	200	Success	61.0
Aug 11, 2015 2:39:09 AM	OmniPortlet	wc-OmniPortlet	Web	200	Success	31.0
Aug 11, 2015 2:38:50 AM	OmniPortlet	wc-OmniPortlet	Web	500	Failure	30001.0
Aug 11, 2015 2:38:42 AM	OmniPortlet	wc-OmniPortlet	Web	500	Failure	30001.0
Aug 11, 2015 2:38:12 AM	Parameter Form	wc-WSRPTools	WSRP	500	Failure	30014.0
Aug 11, 2015 2:37:42 AM	SimpleParameterForm	wc-OmniPortlet	Web	200	Success	8.0

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 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 20-2.

Table 20-5 Recent Portlet Metrics

Metric	Description
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%.

Table 20-5 (Cont.) Recent Portlet Metrics

Metric	Description
Date Time	Date and time of the portlet request.
Portlet Name	Name of the portlet requested.

Overall Portlet Producer Metrics

Historical performance metrics are also available for portlet producers used by WebCenter Portal, as shown in Figure 20-7. The information displayed on this page is described in the following tables:

- Table 20-6
- Table 20-7

> **Note:**
>
> To access these metrics through Fusion Middleware Control, see Viewing Performance Metrics Using Fusion Middleware Control.

Figure 20-7 Portlet Producer Metrics

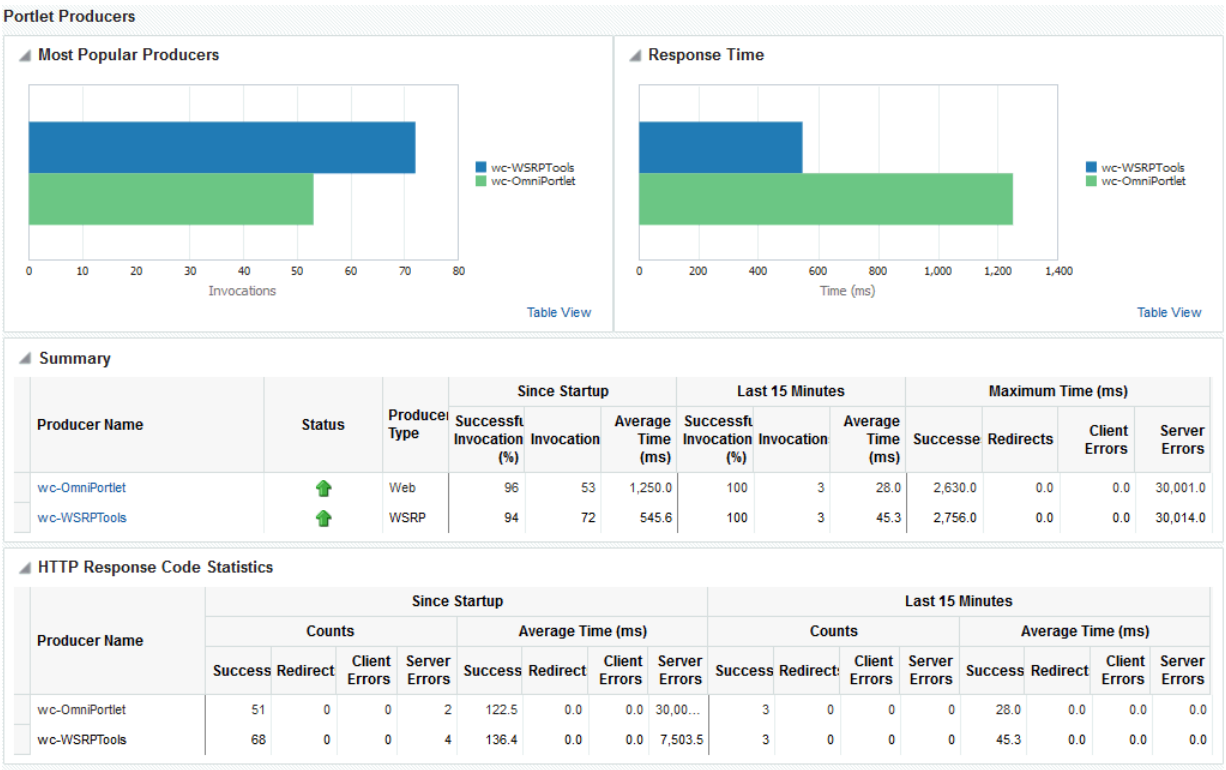


Table 20-6 Portlet Producers - Summary

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 - Last 15 Minutes 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 Viewing and Configuring Log Information.
Invocations	The number of portlet producer invocations per minute: - Since Startup - Last 15 Minutes 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 - Last 15 Minutes

Table 20-7 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.

Table 20-7 (Cont.) Portlet Producer - Detail

Metric	Description
Response Time	The average time each portlet producer takes to process producer requests since WebCenter Portal 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 20-9.
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.
Successful Invocations (%)	The percentage of producer invocations that succeeded: - Since Startup - Last 15 Minutes
Invocations	The number of invocations, per producer: - Since Startup - Last 15 Minutes By sorting the table on this column, you can find the most frequently accessed portlet producer in WebCenter Portal.
Average Time (ms)	The average time taken to make a portlet request, regardless of the result: - Since Startup - Last 15 Minutes 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.

Table 20-7 (Cont.) Portlet Producer - Detail

Metric	Description
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

Overall Portlet Metrics

Historical performance metrics are available for individual portlets used by WebCenter Portal, as shown in Figure 20-8. The information displayed on this page is described in the following tables:

- Table 20-8
- Table 20-9
- Table 20-10
- Table 20-11

> **Note:**
>
> To access these metrics through Fusion Middleware Control, see Viewing Performance Metrics Using Fusion Middleware Control.

Figure 20-8 Portlet Metrics

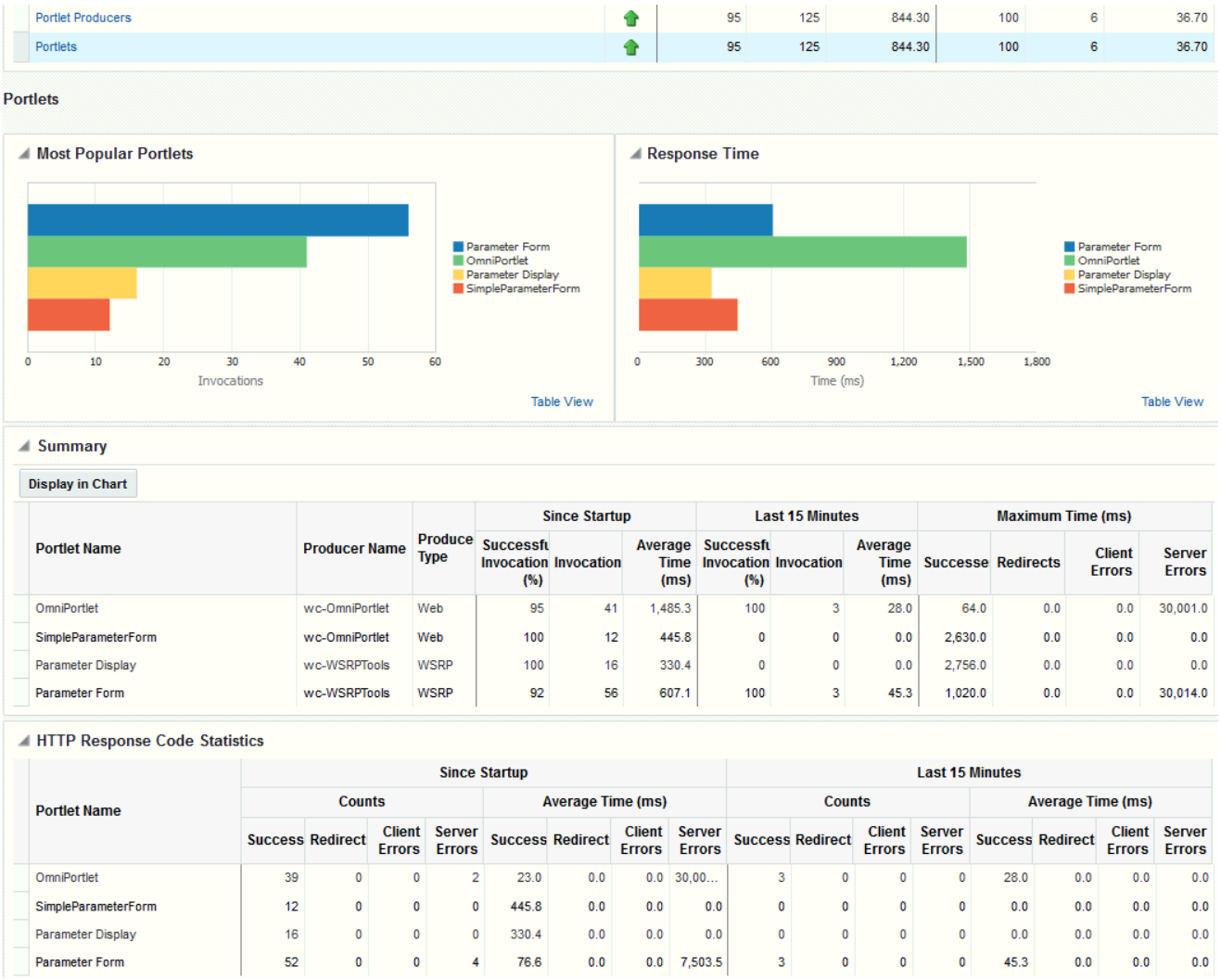


Table 20-8 Portlets - Summary

Metric	Description
Status	The current status of portlets used in WebCenter Portal: • **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 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 20-8 (Cont.) Portlets - Summary

Metric	Description
Successful Invocations (%)	The percentage of portlet invocations that succeeded: - Since Startup - Last 15 Minutes 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 Viewing and Configuring Log Information.
Invocations	The number of portlet invocations per minute: - Since Startup - Last 15 Minutes 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 - Last 15 Minutes

Table 20-9 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 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 20-9 (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 - Last 15 Minutes If Successful Invocations (%) is below 100%, check the diagnostic logs to establish why service requests are failing. See Viewing and Configuring Log Information.
Invocations	The number of invocations, per portlet: - Since Startup - Last 15 Minutes By sorting the table on this column, you can find the most frequently accessed portlet in WebCenter Portal.
Average Time (ms)	The average time each portlet takes to process requests, regardless of the result: - Since Startup - Last 15 Minutes 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 20-10 Portlet - HTTP Response Code Statistics

Metric	Description
Portlet Name	The name of the portlet being monitored.

Table 20-10 (Cont.) Portlet - HTTP Response Code Statistics

Metric	Description
Invocations Count - Successes - Redirects - Client Errors - Server Errors	The number of invocations, by type (HTTP response code): - Since Startup - Last 15 Minutes See Table 20-11.
Average Time (ms) - Successes - Redirects - Client Errors - Server Errors	The average time each portlet takes to process requests: - Since Startup - Last 15 Minutes 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 20-11 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.

Understanding WebLogic Server Metrics

Recent WebLogic Server performance is summarized on the home page for WebCenter Portal (Figure 20-9 and Table 20-12). 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 Navigating to the Home Page for WebCenter Portal.

Figure 20-9 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 Customizing Key Performance Metric Thresholds and Collection.

Table 20-12 Recent WebLogic Server Metrics on the Home Page

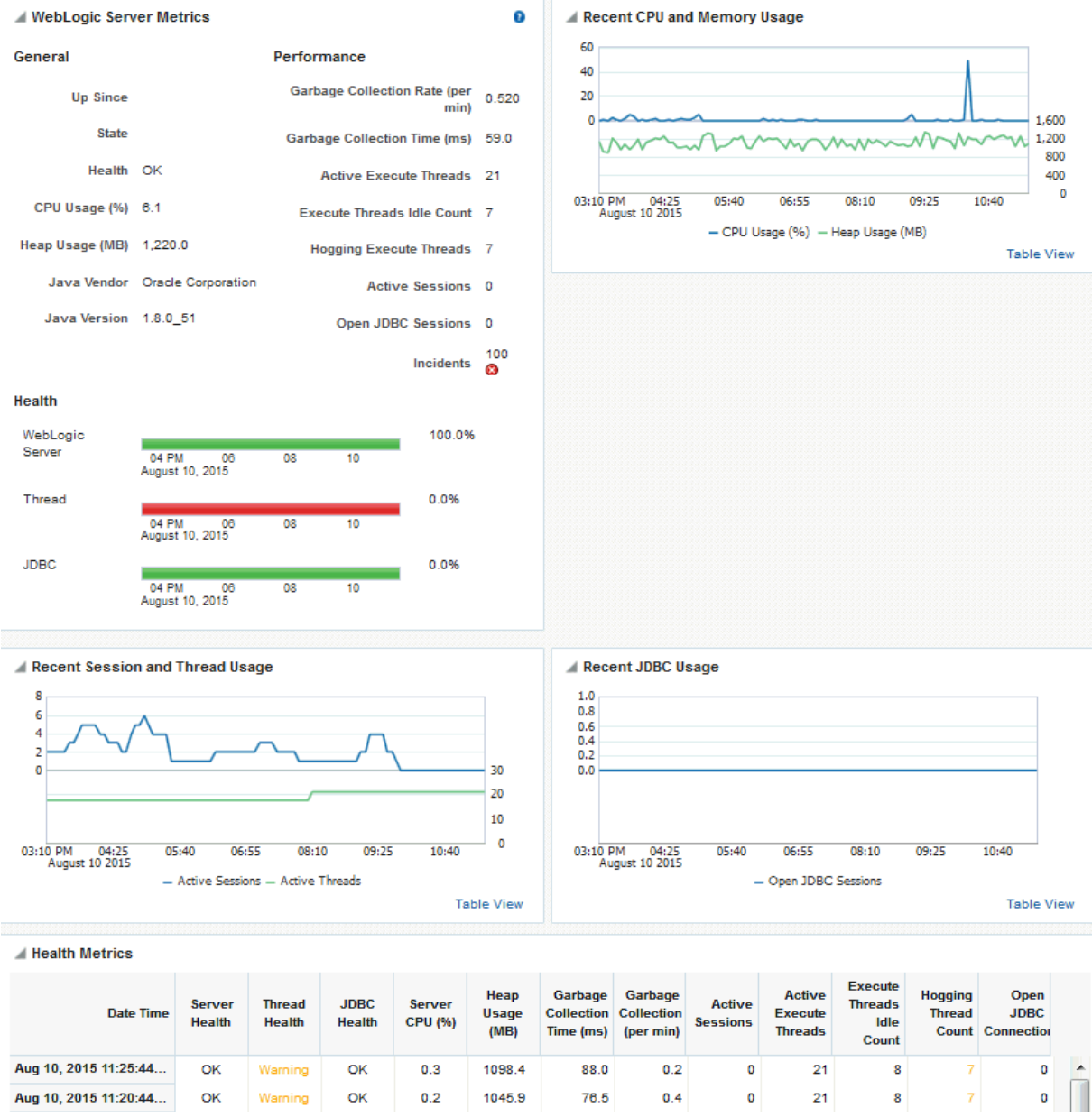
Metric	Description
Health	Summarizes recent WebLogic Server health as reported by the Oracle WebLogic Server self-health monitoring feature. This metric considers recent server health, thread health, and JDBC health: • **Green** - Indicates successful WebLogic Server health checks. • **Red** - Indicates that an incident occurred during a WebLogic Server health check. Click **Health** to identify health checks that fail (do not report OK). See Figure 20-10. • **%** - 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 20-9) 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 20-9). 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 20-10 Recent WebLogic Server Metrics Page



WebLogic Server Metrics Section

Metric	Description
General	

Metric	Description
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.
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.
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 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 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.

Health Metrics Section

This table displays data from the last one hundred (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.

Metric	Description
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 Configure health monitoring in Oracle WebLogic Server Administration Console online help.
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 Configure health monitoring in 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 Configure health monitoring in Oracle WebLogic Server Administration Console online help.
Server CPU (%)	If you are using the Oracle JRockit JDK, this metric shows the 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)	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 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 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.

Metric	Description
Open JDBC Connections	Number of JDBC connections currently open.

Understanding Security Metrics

Some key security-related performance metrics are displayed for WebCenter Portal on the home page (Figure 20-11 and Table 20-13).

> **Note:**
>
> To access the home page, see Navigating to the Home Page for WebCenter Portal.

Figure 20-11 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 20-13 Security Metrics

Metric	Description
LDAP Cache Hit Ratio (%)	Percentage of LDAP searches that result in a cache hit.
Average LDAP Lookup Time (ms)	Average time to complete an LDAP search request: - Since Startup - Last 15 Minutes[1] If LDAP searches are taking too long, its most likely an issue on the LDAP server that is causing slow response times.

1 The last 10-15 minutes of data is used to calculate recent performance metrics. For details, seeUnderstanding Oracle WebCenter Portal Metric Collection .

Understanding Page Response and Load Metrics

The page response chart on your application's home page (Figure 20-11) shows you how quickly WebLogic Server is responding to page requests and how many requests are being processed (its load).

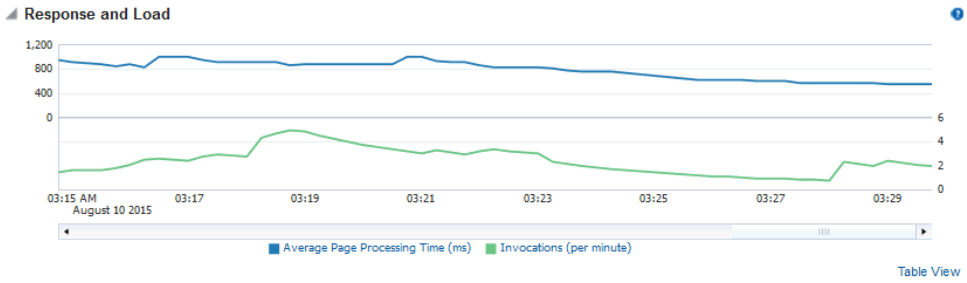
The average page processing time (in ms) for all portals, is calculated over a 15 minute period. The number of invocations per minute is also displayed to help you determine whether the average page processing time is increasing or decreasing. If slower page processing times are due to a large number 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 Navigating to the Home Page for WebCenter Portal.

Figure 20-12 Page Response Metrics on the Home Page



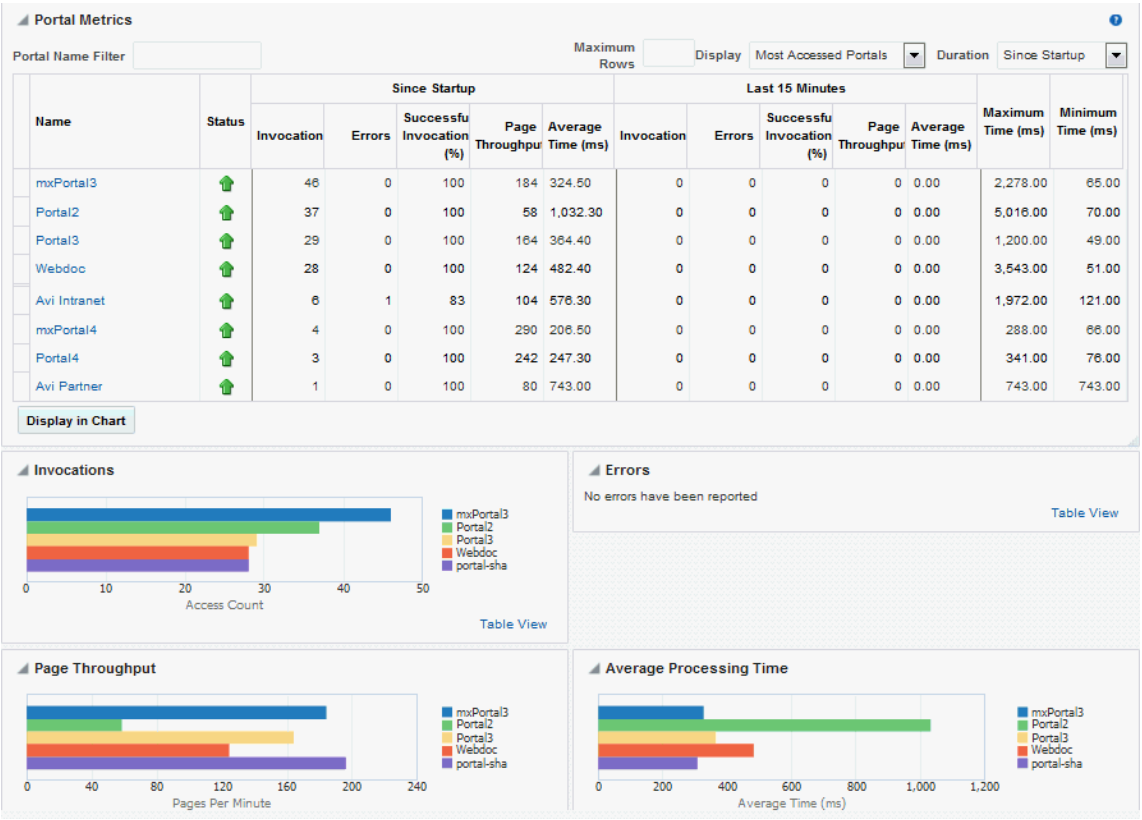
If you compare **Since Startup** metrics with **Recent History** metrics (last 15 minutes), you can determine whether performance has recently deteriorated, and if so, by how much.

Understanding Portal Metrics

(WebCenter Portal only) You can view live performance metrics for individual portals through Fusion Middleware Control, as shown in Figure 20-13. The metrics displayed on this page are described in Table 20-14 and Metrics Common to all Tools and Services.

> **Note:**
>
> Metrics for the Home portal are not included.

Figure 20-13 Portal Metrics



To monitor these metrics through Fusion Middleware Control, see 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:

- **Portal Name Filter** - Enter a full or partial search term, and then press Enter to refresh the list with all portals for which a match is found in the display name. To display metrics for all portals, clear the search term and press Enter again.
- **Maximum Rows** - Restrict the total number of portals displayed in the table.
- **Display** - Display metrics for the most accessed 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 20-14 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.
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 - Last 15 Minutes
Errors	Number of errors recorded.
Successful Invocations (%)	Percentage of portal invocations that succeeded: - Since Startup - Last 15 Minutes If Successful Invocations (%) is below 100%, check the diagnostic logs to establish why portal requests are failing. See Viewing and Configuring Log Information.
Page Throughput	The average number of pages processed per minute for each portal: - Since Startup - Last 15 Minutes
Average Time (ms)	The average time (in ms) to display pages in the portal: - Since Startup - Last 15 Minutes
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.

Portal Metrics - Graphs

Use the graphs below the table to see information about portals:

- **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**.

Understanding Tool and Service Metrics

This section includes the following topics:

- Metrics Common to all Tools and Services
- Metrics Specific to a Particular Tool or Service
- Troubleshooting Common Issues with Tools and Services

Metrics Common to all Tools and Services

Fusion Middleware Control provides capabilities to monitor performance of tools and services used in WebCenter Portal in the following ways:

- Services summary: Summary of performance metrics for each tool or service used in WebCenter Portal. Table 20-15 lists tools and services that use common performance metrics and Table 20-16 describes the common metrics.
- Most popular operations and response time for individual operations. Table 20-17 describes these metrics.
- Per operation metrics: Performance metrics for individual operations. Table 20-15 lists common performance metrics used to monitor performance of individual operations. Table 20-17 describes these metrics.

Table 20-15 Common Metrics for Tools and Services

Tool or Service	Services Summary (Since Startup and Last 15 Minutes)	Per Operation Metrics (Since Startup and Last 15 Minutes)
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)
SOA Server	The performance metrics include: • Status • Successful Invocations (%) • Invocations • Average Time (ms)	Not applicable

Table 20-15 (Cont.) Common Metrics for Tools and Services

Tool or Service	Services Summary (Since Startup and Last 15 Minutes)	Per Operation Metrics (Since Startup and Last 15 Minutes)
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)
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)

Table 20-15 (Cont.) Common Metrics for Tools and Services

Tool or Service	Services Summary (Since Startup and Last 15 Minutes)	Per Operation Metrics (Since Startup and Last 15 Minutes)
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)
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)

Table 20-15 (Cont.) Common Metrics for Tools and Services

Tool or Service	Services Summary (Since Startup and Last 15 Minutes)	Per Operation Metrics (Since Startup and Last 15 Minutes)
RSS	The performance metrics include: • Status • Successful Invocations (%) • Invocations • Average Time (ms)	Not available
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 20-16 describes metrics used for monitoring performance of all operations.

Table 20-16 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 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 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 - Last 15 Minutes If Successful Invocations (%) is below 100%, check the diagnostic logs to establish why service requests are failing. See Viewing and Configuring Log Information.

Table 20-16 (Cont.) Description of Common Metrics - Summary (All Operations)

Metric	Description
Invocations	Number of service invocations per minute: - Since Startup - Last 15 Minutes 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.
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 - Last 15 Minutes 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 20-17 describes metrics used to monitor performance of each operation performed by a tool, service or component.

Table 20-17 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 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 Metrics Specific to a Particular Tool or Service.
Invocations	The number of invocations, per operation: - Since Startup - Last 15 Minutes 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.

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:

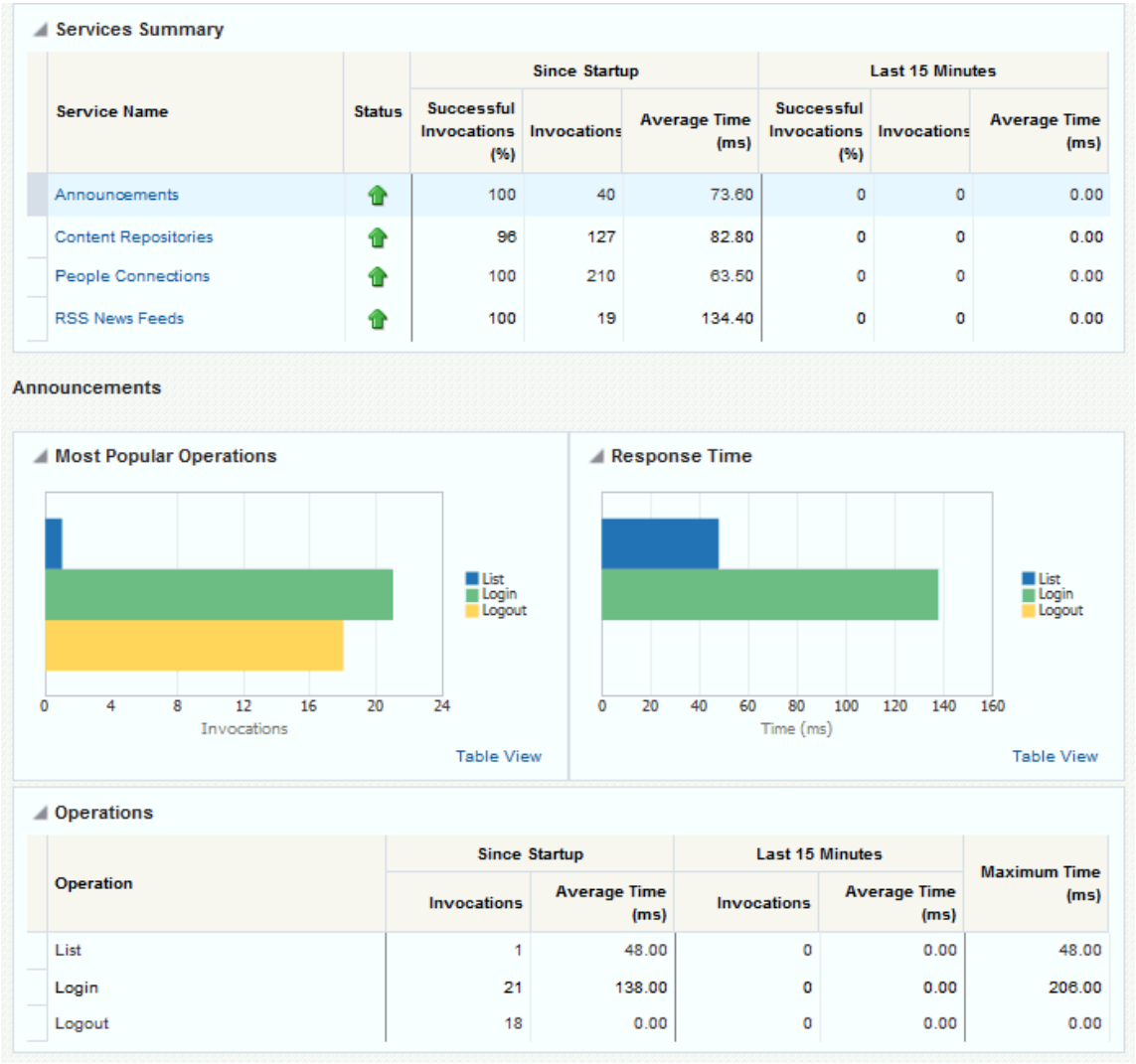
- Announcements Metrics
- BPEL Worklist Metrics
- Content Repository Metrics
- Discussion Metrics
- Events Metrics
- External Application Metrics
- Instant Messaging and Presence Metrics
- Import and Export Metrics
- List Metrics
- Mail Metrics
- Note Metrics
- Page Operation Metrics
- People Connection Metrics
- RSS News Feed Metrics
- Search Metrics

To access live performance metrics for WebCenter Portal, see Viewing Performance Metrics Using Fusion Middleware Control.

Announcements Metrics

Performance metrics associated with announcements (Figure 20-14) are described in Table 20-18 and Metrics Common to all Tools and Services.

Figure 20-14 Announcements Metrics



To monitor these metrics through Fusion Middleware Control, see Viewing Performance Metrics Using Fusion Middleware Control.

Table 20-18 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 Announcements - Issues and Actions. For common causes, see 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 Announcements - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.

Table 20-18 (Cont.) Announcements - Operations Monitored

Operation	Description	Performance Issues - User Action
Search	Searches for terms within announcement text.	If announcement searches are failing, verify that announcement text contains the search terms. For other causes, see Announcements - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Create	Creates an announcement.	For specific causes, see Announcements - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
List	Retrieves a list of announcements.	For specific causes, see Announcements - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.

BPEL Worklist Metrics

Performance metrics associated with worklists are described in Metrics Common to all Tools and Services.

To monitor these metrics through Fusion Middleware Control, see Viewing Performance Metrics Using Fusion Middleware Control.

Content Repository Metrics

Performance metrics associated with documents and Content Presenter (Figure 20-15 and Figure 20-16) are described in the following tables:

- Table 20-19
- Table 20-20
- Table 20-21
- Table 20-22

Figure 20-15 Content Repository Metrics

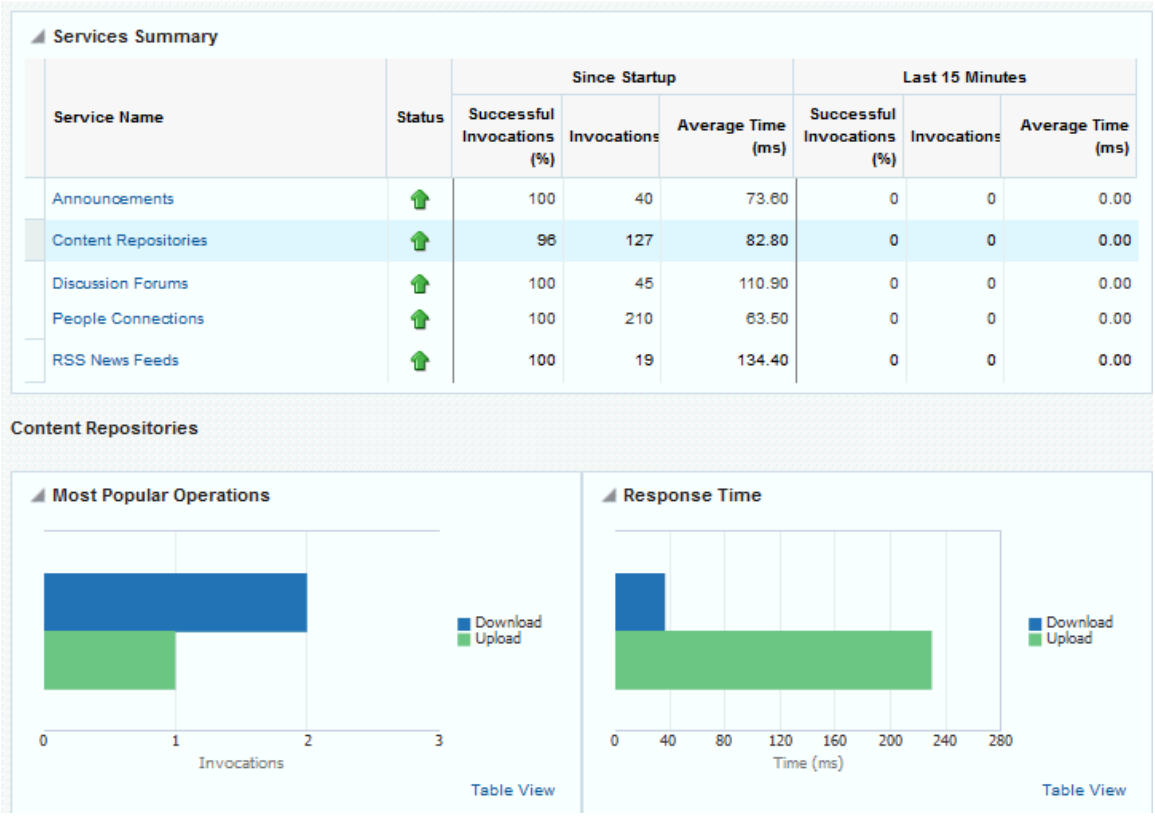
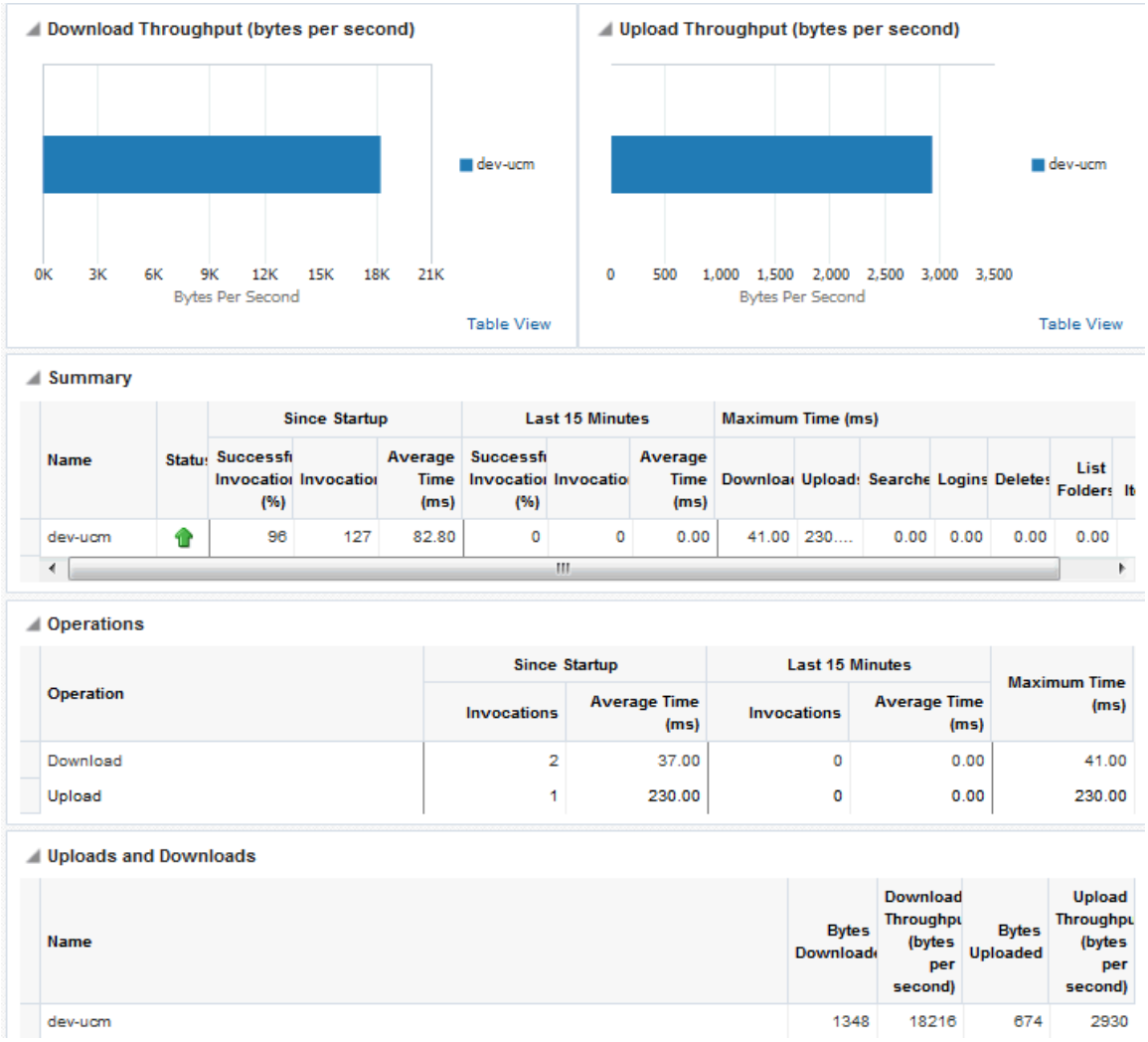


Figure 20-16 Content Repository Metrics - Per Operation



To monitor these metrics through Fusion Middleware Control, see Viewing Performance Metrics Using Fusion Middleware Control.

Table 20-19 Content Repository - Operations Monitored

Operation	Description	Performance Issues - User Action
Download	Downloads one or more documents from a content repository.	For specific causes, see Content Repository (Documents and Content Presenter) - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.

Table 20-19 (Cont.) Content Repository - Operations Monitored

Operation	Description	Performance Issues - User Action
Upload	Uploads one or more documents to a content repository.	For specific causes, see Content Repository (Documents and Content Presenter) - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Search	Searches for documents stored in a content repository.	For specific causes, see Content Repository (Documents and Content Presenter) - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Login	Establishes a connection to the content repository and authenticates the user.	For specific causes, see Content Repository (Documents and Content Presenter) - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Delete	Deletes one or more documents stored in a content repository.	For specific causes, see Content Repository (Documents and Content Presenter) - Issues and Actions. For common causes, see 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 Content Repository (Documents and Content Presenter) - Issues and Actions. For common causes, see 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 Content Repository (Documents and Content Presenter) - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.

Table 20-20 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 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.
Successful Invocations (%)	The percentage of document invocations that succeeded (Upload, Download, Search, Login, Delete): - Since Startup - Last 15 Minutes If Successful Invocations (%) is below 100%, check the diagnostic logs to establish why service requests are failing. See Viewing and Configuring Log Information.
Invocations	The number of document invocations per minute (Upload, Download, Search, Login, Delete): - Since Startup - Last 15 Minutes 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 - Last 15 Minutes
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.

Table 20-20 (Cont.) Content Repository Metrics - Summary (All Repositories)

Metric	Description
Response Time	The average time to process operations associated with documents since WebCenter Portal 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 20-21 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 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 - Last 15 minutes If Successful Invocations (%) is below 100%, check the diagnostic logs to establish why service requests are failing. See Viewing and Configuring Log Information.

Table 20-21 (Cont.) Content Repository Metrics - Operation Summary Per Repository

Metric	Description
Invocations	The number of document invocations per minute (Upload, Download, Search, Login, Delete) for this content repository: - Since Startup - Last 15 minutes 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 - Last 15 minutes
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.
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.

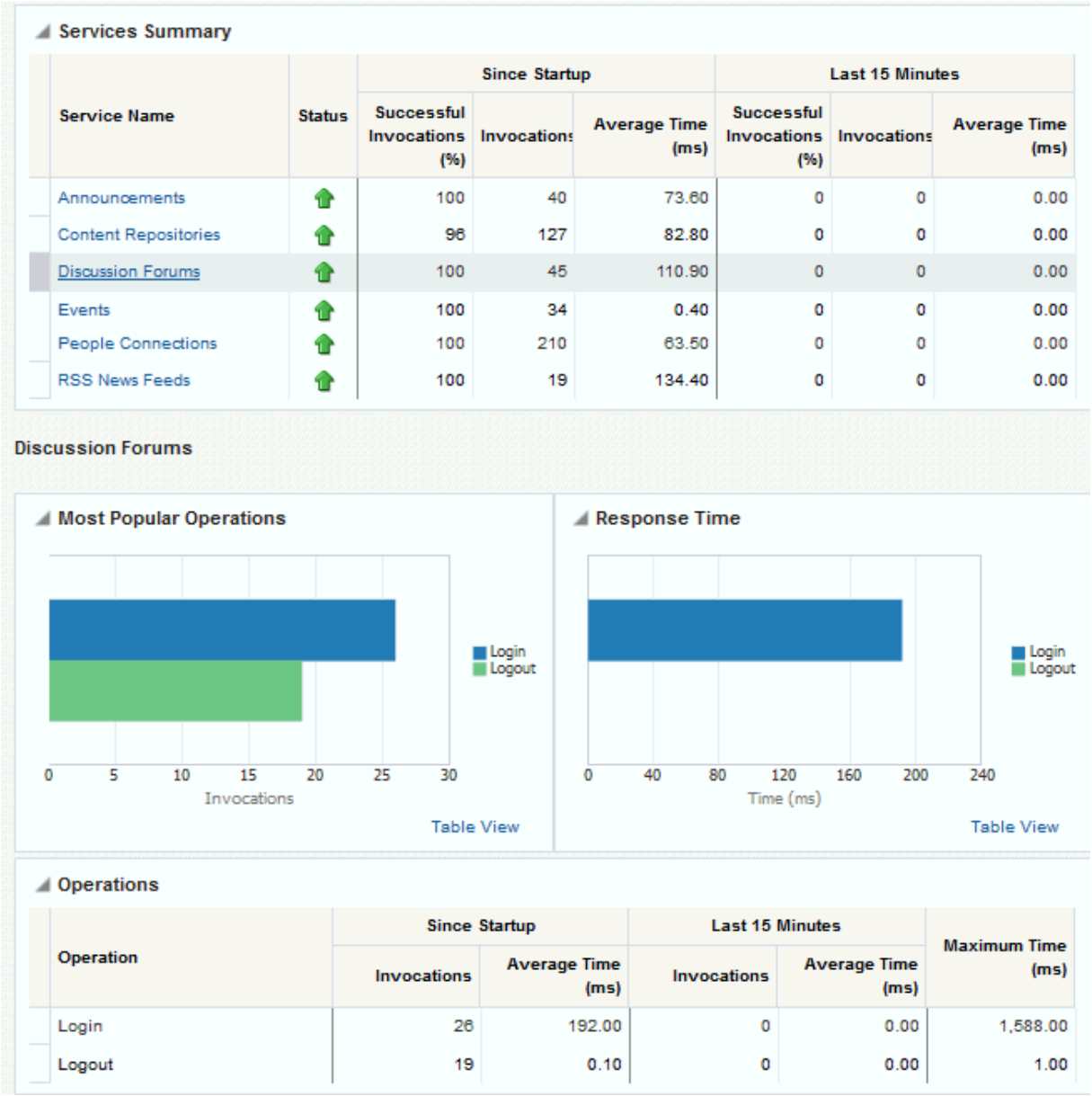
Table 20-22 Content Repository Metrics - Operation Detail Per Repository

Metric	Description
Invocations	The number of invocations per document operation (Upload, Download, Search, Login, Delete): - Since Startup - Last 15 minutes 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 - Last 15 minutes

Discussion Metrics

Performance metrics associated with discussions (Figure 20-17) are described in Table 20-23 and Metrics Common to all Tools and Services.

Figure 20-17 Discussion Metrics



To monitor these metrics through Fusion Middleware Control, see Viewing Performance Metrics Using Fusion Middleware Control.

Table 20-23 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 Discussions - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.

Table 20-23 (Cont.) Discussions - Operations Monitored

Operation	Description	Performance Issues - User Action
Logout	Logs a WebCenter Portal user out of the discussions server that is hosting discussion forums.	For specific causes, see Discussions - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
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 Discussions - Issues and Actions. For common causes, see 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 Discussions - Issues and Actions. For information on common causes, see 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 Discussions - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
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 Discussions - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.

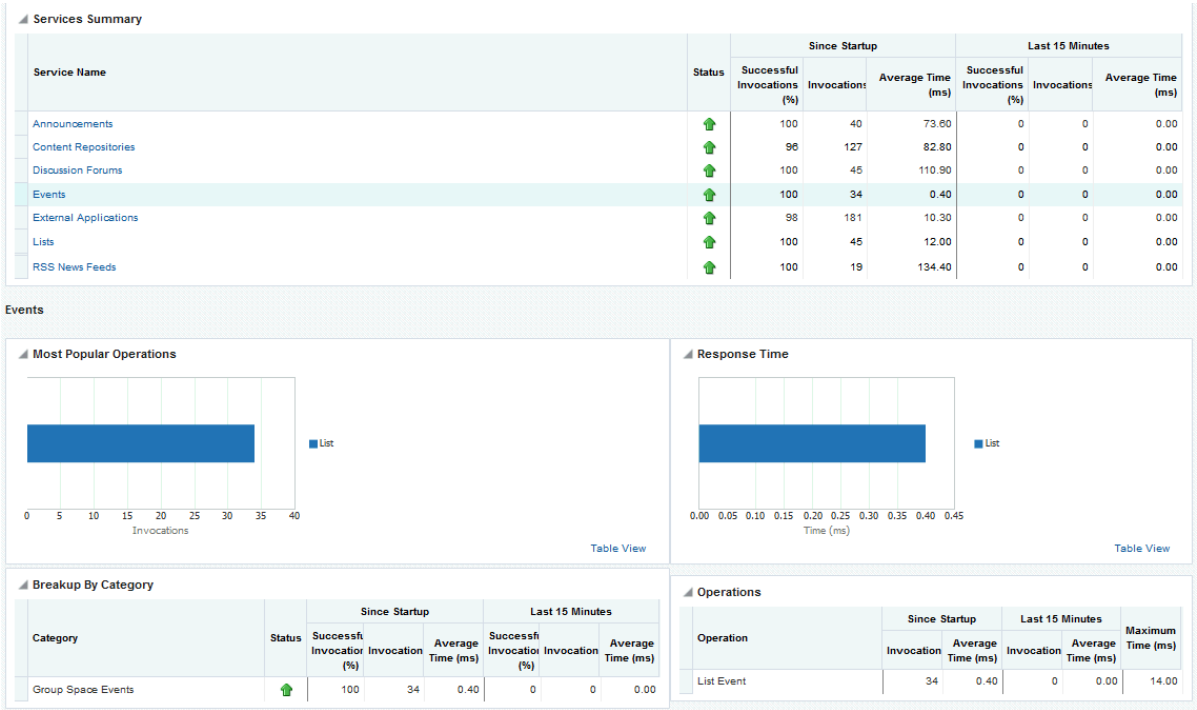
Table 20-23 (Cont.) Discussions - Operations Monitored

Operation	Description	Performance Issues - User Action
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 Discussions - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.

Events Metrics

Performance metrics associated with events are described in Table 20-24 and Metrics Common to all Tools and Services.

Figure 20-18 Events Metrics



To monitor these metrics through Fusion Middleware Control, see Viewing Performance Metrics Using Fusion Middleware Control.

Table 20-24 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 Events - Issues and Actions. For common causes, see 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 Events - Issues and Actions. For common causes, see 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 Events - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
List Event	Retrieves a list of events from the WebCenter Portal's repository.	For specific causes, see Events - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Search Event	Searches for terms within event text.	For specific causes, see Events - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.

External Application Metrics

Performance metrics associated with external applications are described in Table 20-25 and Metrics Common to all Tools and Services.

Figure 20-19 External Application Metrics

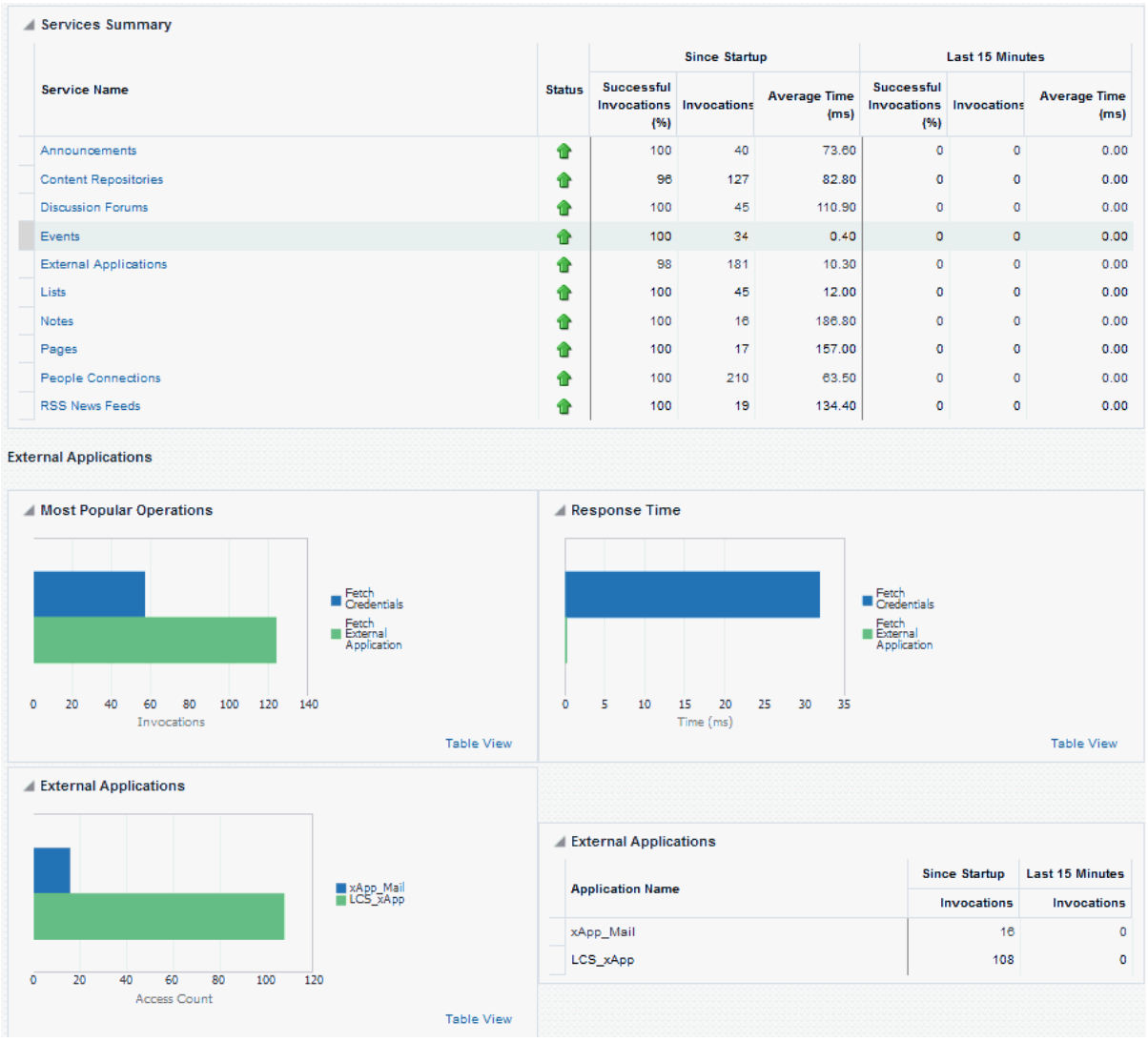
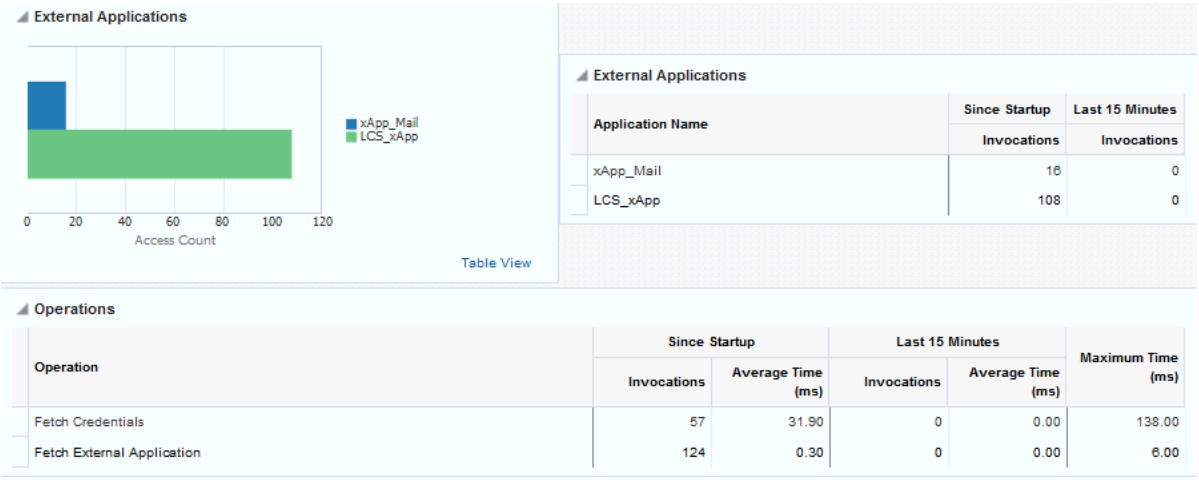


Figure 20-20 External Application Metrics - Per Operation



To monitor these metrics through Fusion Middleware Control, see Viewing Performance Metrics Using Fusion Middleware Control.

Table 20-25 External Applications - Operations Monitored

Operation	Description	Performance Issues - User Action
Fetch Credentials	Retrieves credentials for an external application.	For specific causes, see External Applications - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Store Credentials	Stores user credentials for an external application.	For specific causes, see External Applications - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Fetch External Application	Retrieves an external application.	For specific causes, see External Applications - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Automated Logins	Logs a WebCenter Portal user in to an external application (using the automated login feature).	For specific causes, see External Applications - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.

Instant Messaging and Presence Metrics

Performance metrics associated with instant messaging and presence are described in Table 20-26 and Metrics Common to all Tools and Services.

To monitor these metrics through Fusion Middleware Control, see Viewing Performance Metrics Using Fusion Middleware Control.

Table 20-26 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 Instant Messaging and Presence - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Login	Logs a WebCenter Portal user (accessing the instant messaging and presence) into the instant messaging and presence server.	For specific causes, see Instant Messaging and Presence - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.

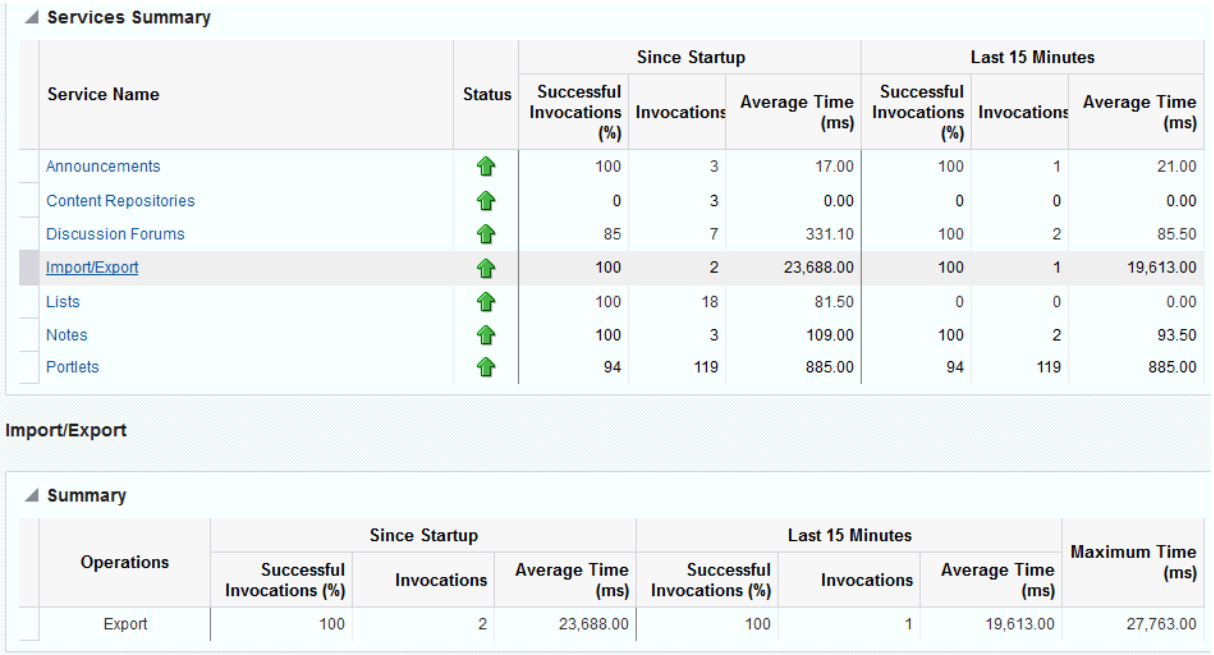
Table 20-26 (Cont.) Instant Messaging and Presence - Operations Monitored

Operation	Description	Performance Issues - User Action
Logout	Logs a WebCenter Portal user (accessing instant messaging and presence) out of the instant messaging and presence server.	For specific causes, see Instant Messaging and Presence - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.

Import and Export Metrics

Performance metrics associated with import and export (Figure 20-21) are described in Table 20-27 and Metrics Common to all Tools and Services. These metrics apply to WebCenter Portal only.

Figure 20-21 Import/Export Metrics



To monitor these metrics through Fusion Middleware Control, see Viewing Performance Metrics Using Fusion Middleware Control.

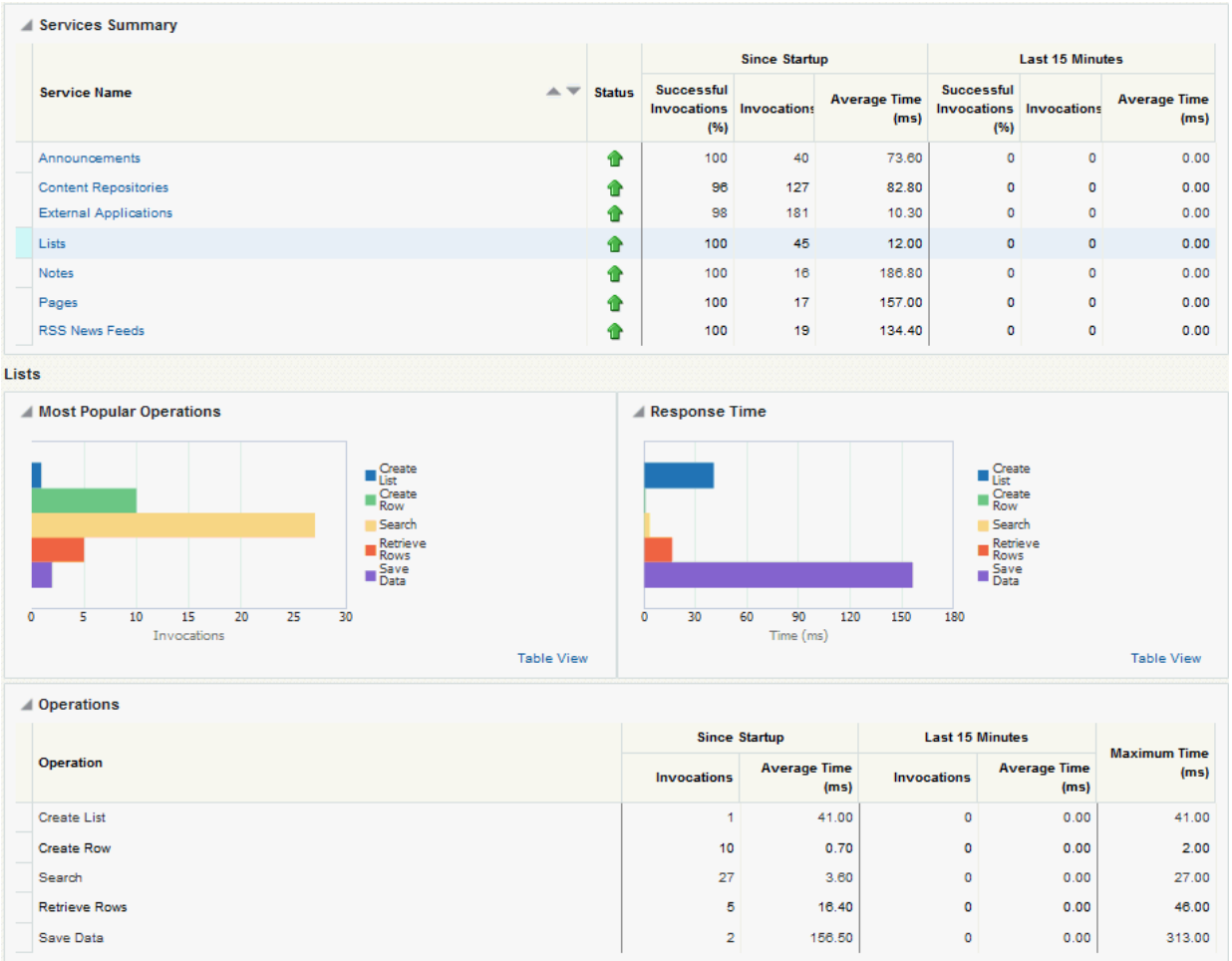
Table 20-27 Import/Export - Operations Monitored

Operation	Description	Performance Issues - User Action
Export	Exports an entireWebCenter Portal application.	For specific causes, see Import and Export - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Import	Imports an entire WebCenter Portal application.	For specific causes, see Import and Export - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.

List Metrics

(WebCenter Portal only) Performance metrics associated with lists (Figure 20-22) are described in Table 20-28 and Metrics Common to all Tools and Services.

Figure 20-22 List Metrics



To monitor these metrics through Fusion Middleware Control, see Viewing Performance Metrics Using Fusion Middleware Control.

Table 20-28 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 Lists - Issues and Actions. For common causes, see 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 Lists - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
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 Lists - Issues and Actions. For common causes, see 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 Lists - Issues and Actions. For common causes, see 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 Lists - Issues and Actions. For common causes, see 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 Lists - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Search	Retrieves a list by its ID from the Metadata repository.	For specific causes, see Lists - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.

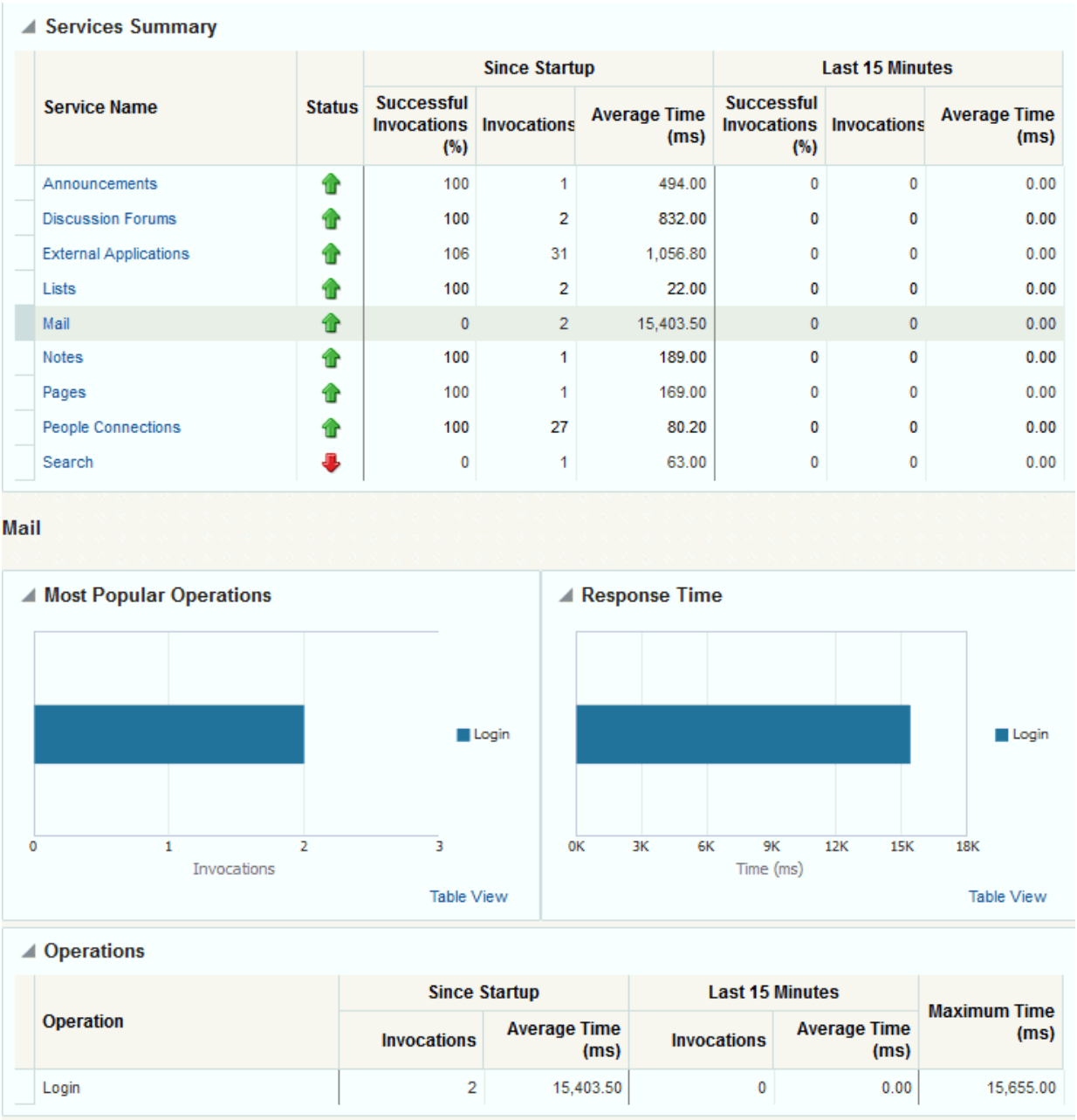
Table 20-28 (Cont.) Lists- Operations Monitored

Operation	Description	Performance Issues - User Action
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 Lists - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.

Mail Metrics

Performance metrics associated with mail (Figure 20-23) are described in Table 20-29 and Metrics Common to all Tools and Services.

Figure 20-23 Mail Metrics



To monitor these metrics through Fusion Middleware Control, see Viewing Performance Metrics Using Fusion Middleware Control.

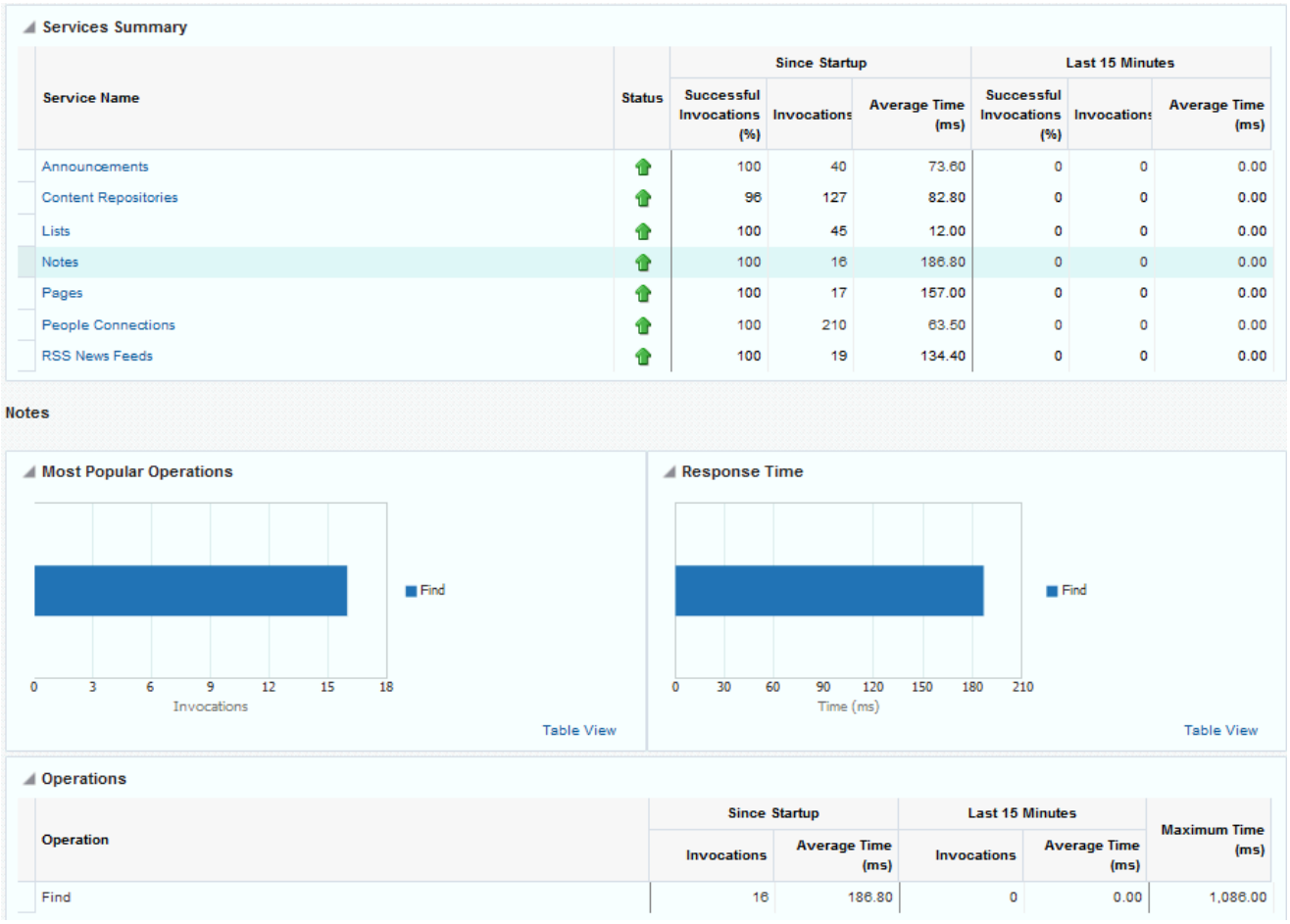
Table 20-29 Mail - Operations Monitored

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 Mail - Issues and Actions. For common causes, see 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 Mail - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Receive	Receives a mail.	For specific causes, see Mail - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Send	Sends a mail.	For specific causes, see Mail - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Search	Searches for mail that contains a specific term.	For specific causes, see Mail - Issues and Actions. For information on common causes, see Understanding Some Common Performance Issues and Actions.

Note Metrics

Performance metrics associated with notes (Figure 20-24) are described in Table 20-30 and Metrics Common to all Tools and Services.

Figure 20-24 Notes Metrics



To monitor these metrics through Fusion Middleware Control, see Viewing Performance Metrics Using Fusion Middleware Control.

Table 20-30 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 Notes - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Update	Updates a personal note. The Save Changes operation commits note updates to the MDS repository.	For specific causes, see Notes - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.

Table 20-30 (Cont.) Notes - Operations Monitored

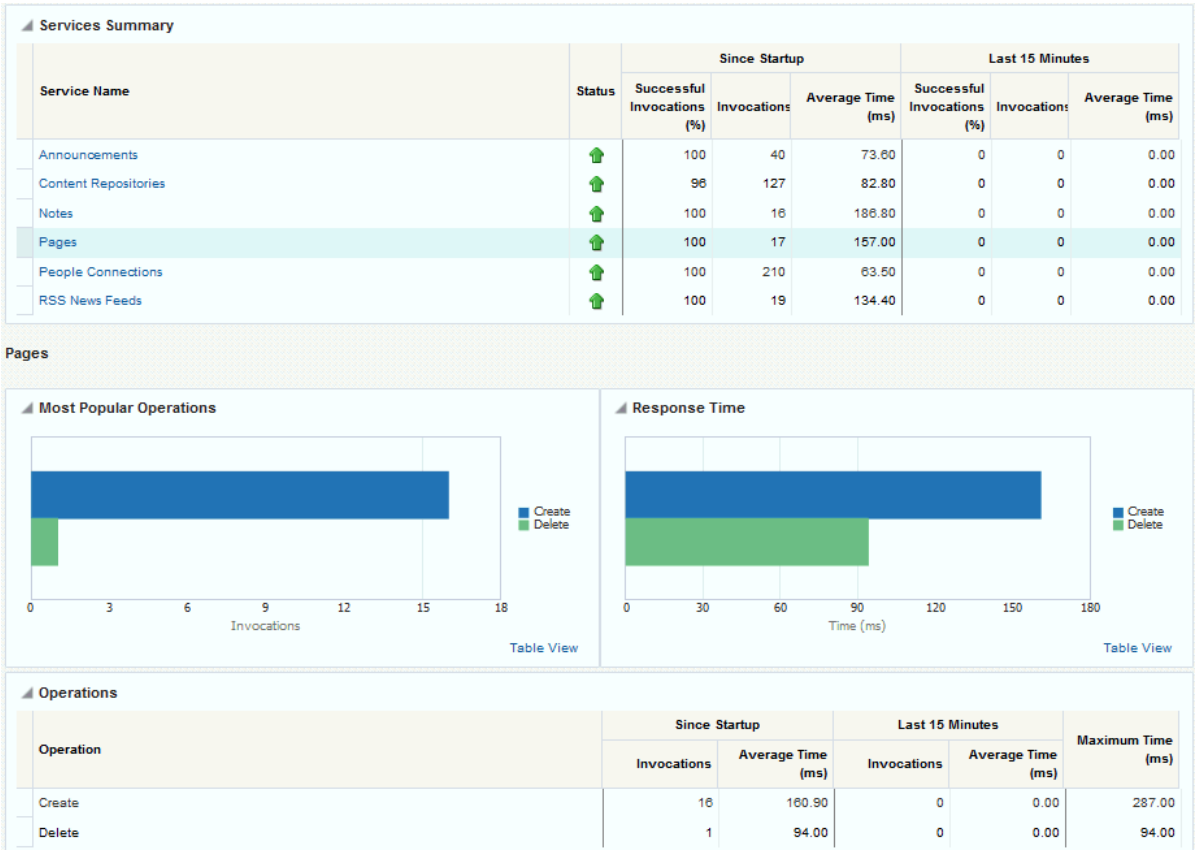
Operation	Description	Performance Issues - User Action
Find	Retrieves a note from the MDS repository.	For specific causes, see Notes - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Delete	Deletes a note from the MDS repository.	For specific causes, see Notes - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.

Page Operation Metrics

Performance metrics associated with the page operations (Figure 20-25) are described in Table 20-31 and 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 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 20-25 Page Operation Metrics



To monitor these metrics through Fusion Middleware Control, see Viewing Performance Metrics Using Fusion Middleware Control.

Table 20-31 Page Service - Operations Monitored

Operation	Description	Performance Issues - User Action
Create	Creates a page in WebCenter Portal.	For specific causes, see Page Services - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Copy	Copies a page.	For specific causes, see Page Services - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Delete	Deletes a page.	For specific causes, see Page Services - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.

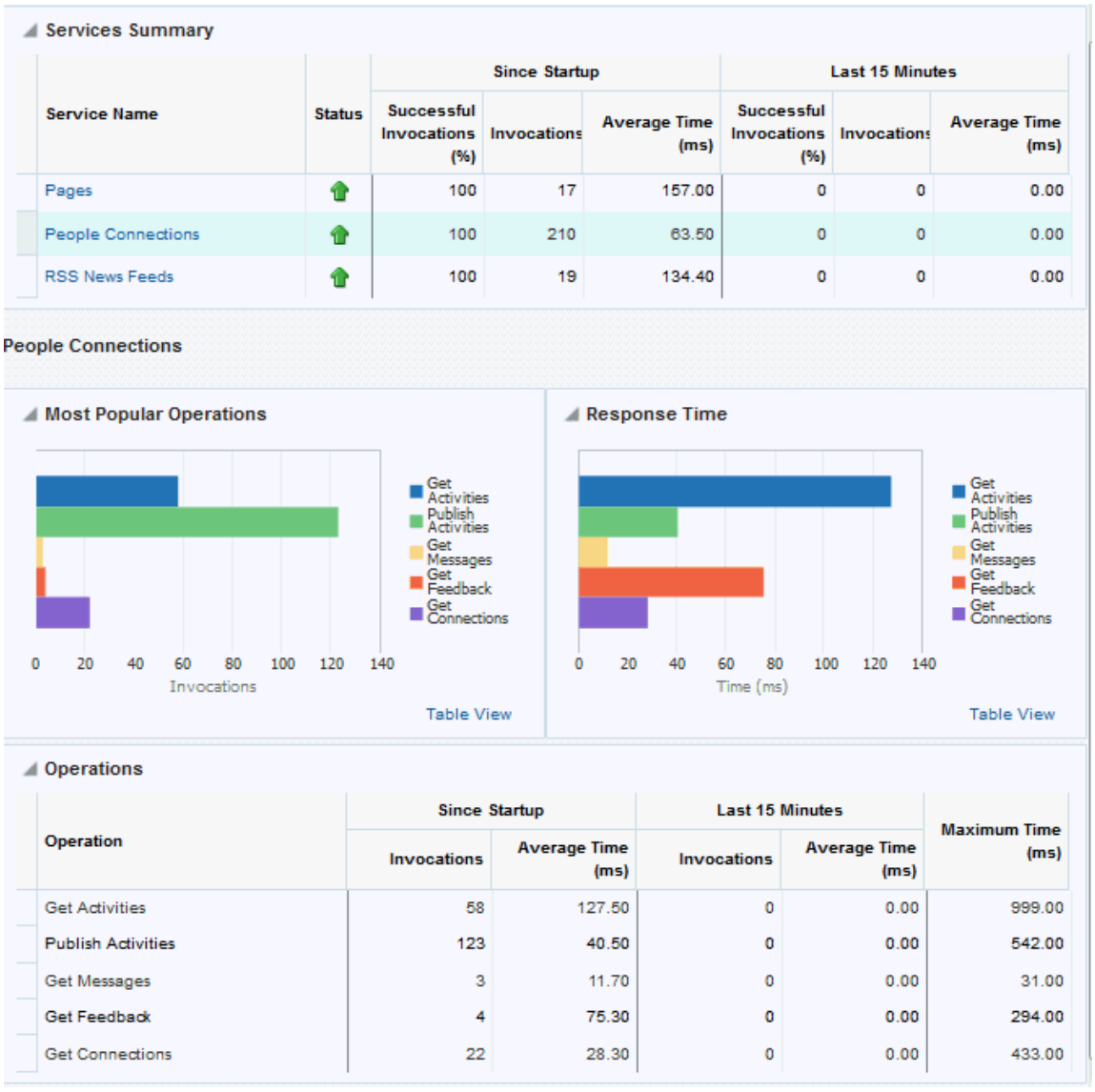
Table 20-31 (Cont.) Page Service - Operations Monitored

Operation	Description	Performance Issues - User Action
Search	Searches for pages that contain a specific term.	For specific causes, see Page Services - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.

People Connection Metrics

Performance metrics associated with people connections are described in Table 20-32 and Metrics Common to all Tools and Services.

Figure 20-26 People Connection Metrics



To monitor these metrics through Fusion Middleware Control, see Viewing Performance Metrics Using Fusion Middleware Control.

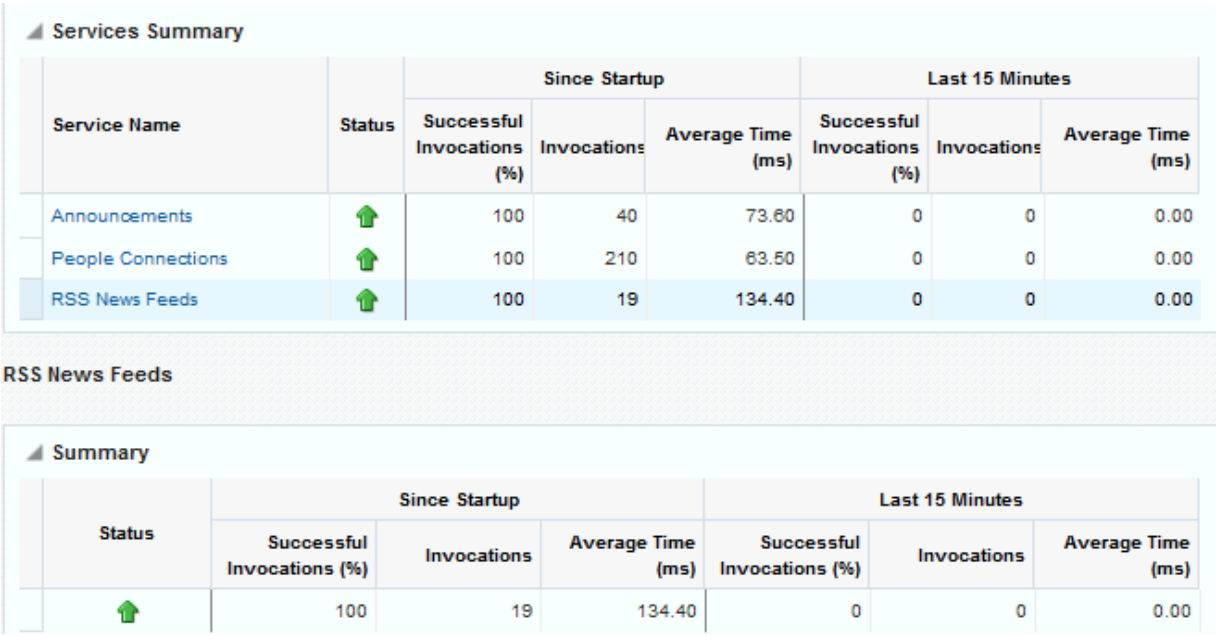
Table 20-32 People Connections - Operations Monitored

Operation	Description	Performance Issues - User Action
Get Profiles	Retrieves profiles of a user.	For specific causes, see People Connections - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Get Activities	Retrieves the activities based on the user filter options.	For specific causes, see People Connections - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Publish Activities	Publishes an activity in the user session and saves it in WebCenter Portal.	For specific causes, see People Connections - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Get Messages	Retrieves the messages of the user.	For specific causes, see People Connections - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Get Feedback	Retrieves the feedback of the user.	For specific causes, see People Connections - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.
Get Connections	Retrieves the connections of users.	For specific causes, see People Connections - Issues and Actions. For common causes, see Understanding Some Common Performance Issues and Actions.

RSS News Feed Metrics

Performance metrics associated with RSS news feeds (Figure 20-27) are described in Metrics Common to all Tools and Services.

Figure 20-27 RSS News Feed Metrics



Services Summary

Service Name	Status	Since Startup			Last 15 Minutes		
		Successful Invocations (%)	Invocations	Average Time (ms)	Successful Invocations (%)	Invocations	Average Time (ms)
Announcements	↑	100	40	73.60	0	0	0.00
People Connections	↑	100	210	63.50	0	0	0.00
RSS News Feeds	↑	100	19	134.40	0	0	0.00

RSS News Feeds

Summary

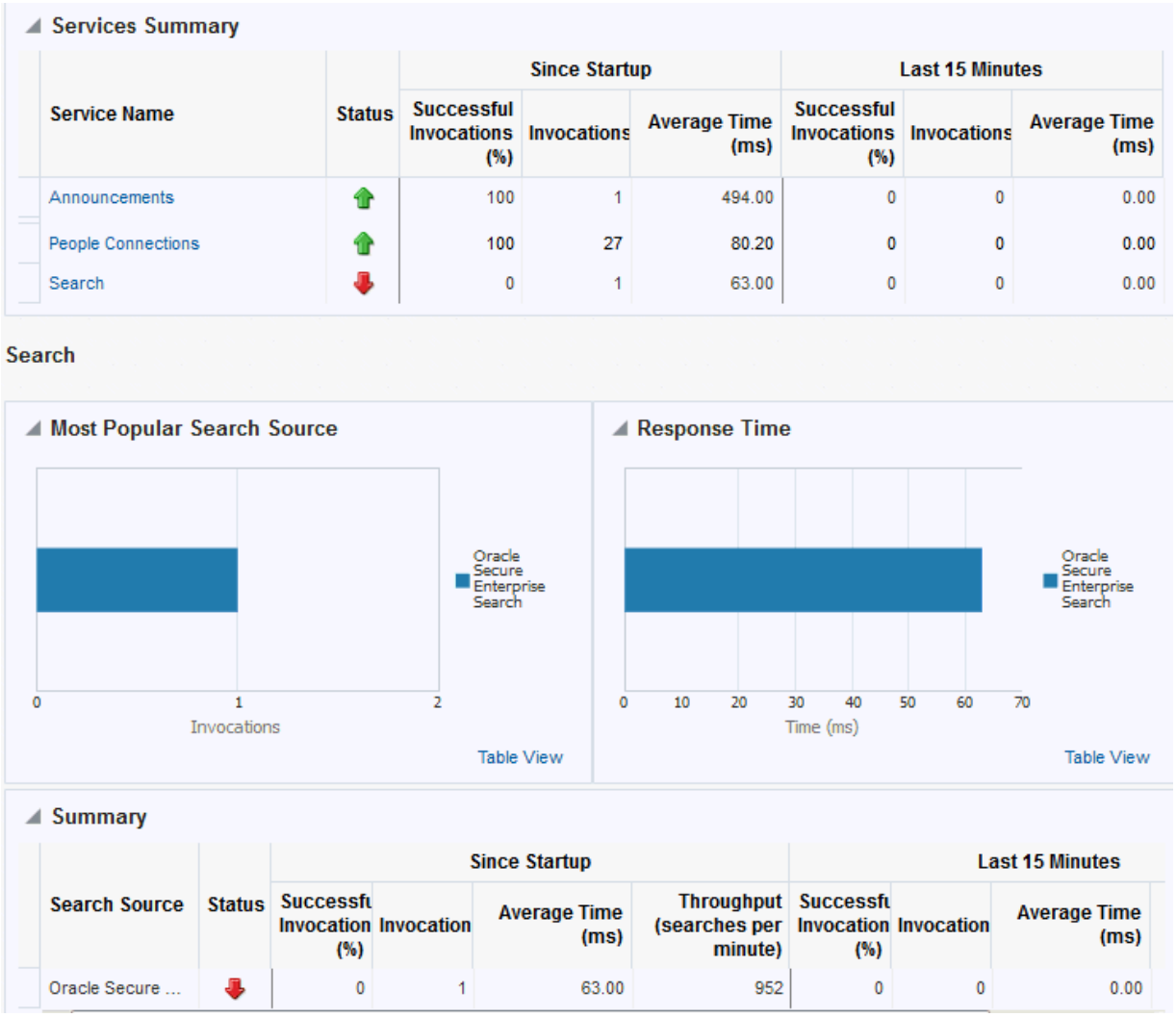
Status	Since Startup			Last 15 Minutes		
	Successful Invocations (%)	Invocations	Average Time (ms)	Successful Invocations (%)	Invocations	Average Time (ms)
↑	100	19	134.40	0	0	0.00

To monitor these metrics through Fusion Middleware Control, see Viewing Performance Metrics Using Fusion Middleware Control.

Search Metrics

Performance metrics associated with search (Figure 20-28) are described in Table 20-33 and Metrics Common to all Tools and Services.

Figure 20-28 Search Metrics



To monitor these metrics through Fusion Middleware Control, see Viewing Performance Metrics Using Fusion Middleware Control.

Table 20-33 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.
Portal Events	Portal 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.

Table 20-33 (Cont.) Search - Search Sources

Operation	Description
Notes	Notes text, such as reminders, is searched.
Elasticsearch	Contents from discussions, tag clouds, notes, and other tools and services are searched.
Pages	Contents added to application, personal, public, wiki, and blog pages are searched.

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:**
>
> Understanding Some Common Performance Issues and Actions

This section includes the following topics:

- Announcements - Issues and Actions
- Content Repository (Documents and Content Presenter) - Issues and Actions
- Discussions - Issues and Actions
- External Applications - Issues and Actions
- Events - Issues and Actions
- Instant Messaging and Presence - Issues and Actions
- Import and Export - Issues and Actions
- Lists - Issues and Actions
- Mail - Issues and Actions
- Notes - Issues and Actions
- Page Services - Issues and Actions
- Portlets and Producers - Issues and Actions
- People Connections - Issues and Actions
- RSS News Feeds - Issues and Actions
- Search - Issues and Actions

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.

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), 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 Using Fusion Middleware Control to Modify Internet Configuration in *Oracle Fusion Middleware Administering Oracle WebCenter Content*.
- (Functional check) Check logs on the back-end server. For Content Server, go to **Content Server** > **Administration** > **Log files** > **Content Server Logs**.
- (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_Portal-diagnostic.log`. See Viewing and Configuring Log Information.

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.

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.

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.

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.

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.

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.

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.

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).

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.

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.

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.

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.

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.

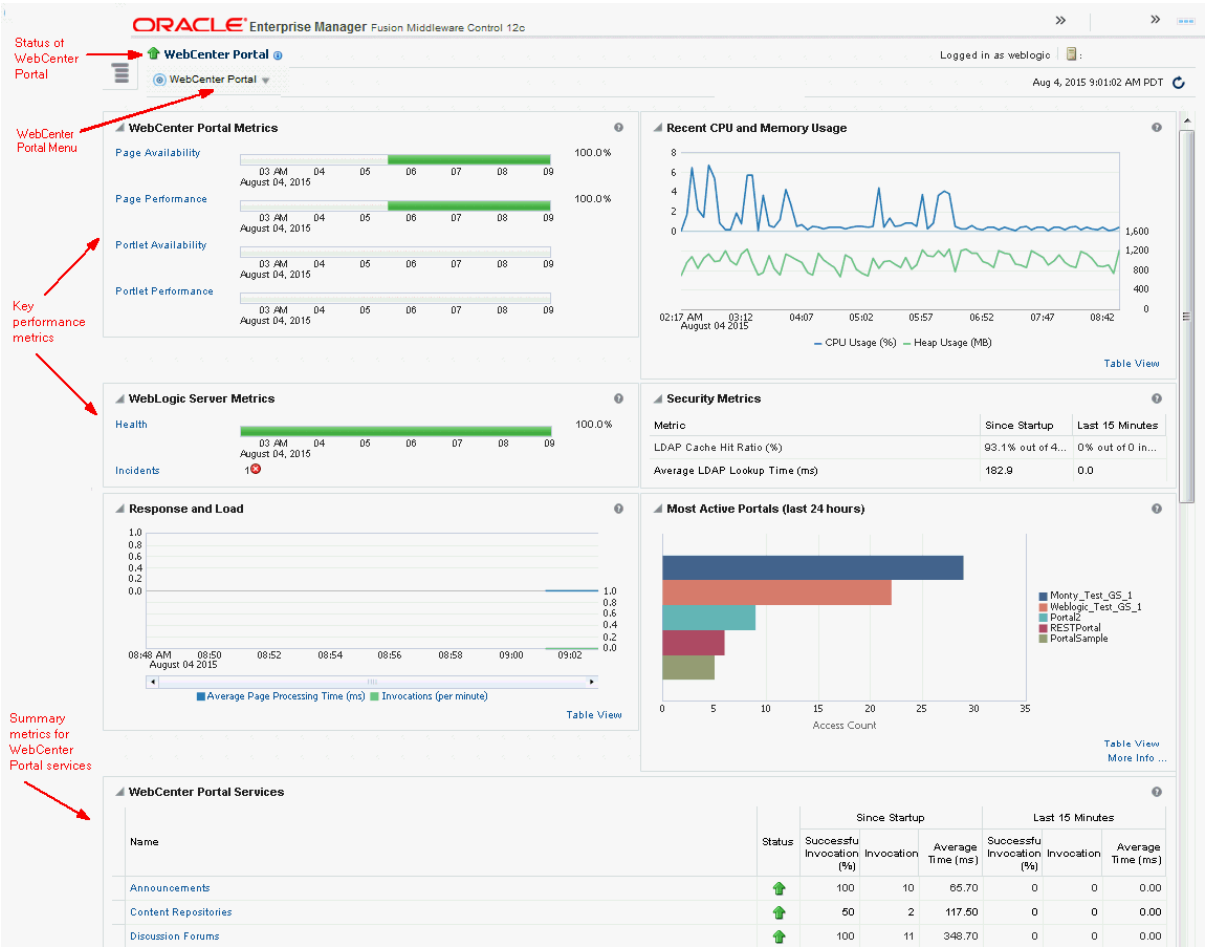
Viewing Performance Metrics Using Fusion Middleware Control

Fusion Middleware Control monitors a wide range of performance metrics for 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 20-29).

Figure 20-29 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 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:

- Monitoring Recent Performance Metrics for WebCenter Portal
- Monitoring Portal Metrics
- Monitoring Page Metrics for WebCenter Portal
- Monitoring Service Metrics for WebCenter Portal
- Monitoring All Metrics Through the Metrics Palette

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 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 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 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 Portlet Producer Metrics, and Understanding WebLogic Server Metrics.

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 Navigating to the Home Page for WebCenter Portal.

2. From the **WebCenter Portal** menu, select **Monitoring** > **Overall Portal Metrics**.

   To learn more about the metrics displayed, see Understanding Portal Metrics. See 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 **Portal Name Filter** field, enter the name of a portal, then press **[Enter]**. For information about portal filtering options, see 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**.

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 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 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 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 Page Operation Metrics.

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 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 Metrics Specific to a Particular Tool or Service. See also, Troubleshooting Common Issues with Tools and Services.

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 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.

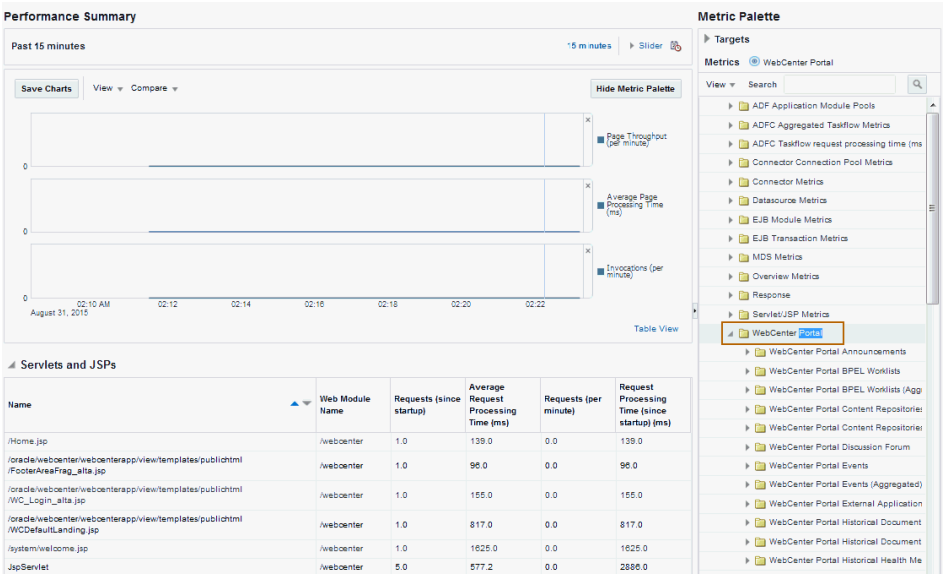
3. In the **Metric Palette**, expand the folders under **WebCenter Portal** and then select the metric check boxes to monitor the metric in graphical or tabular format.

   Figure 20-30 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 20-30 WebCenter Portal - Performance Summary and Metric Palette



Customizing Key Performance Metric Thresholds and Collection

This section includes the following topics:

- Understanding Customization Options for Key Performance Metrics
- Understanding Default Metric Collection and Threshold Settings
- Configuring Thresholds for Key Metrics
- Configuring the Frequency of WebLogic Server Health Checks
- Configuring the Number of Samples Used to Calculate Key Performance Metrics
- Editing Thresholds and Collection Options for WebCenter Portal

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: Configuring Thresholds for Key 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 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 Configuring the Frequency of WebLogic Server Health Checks.

See also, Editing Thresholds and Collection Options for WebCenter Portal.

Understanding Default Metric Collection and Threshold Settings

You can configure metric collection options and metric threshold settings for WebCenter Portal through the `metric_properties.xml` file. The default settings are shown in Example 20-1 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 20-1 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 20-35
- Table 20-36

For information on how to modify the default settings, see Customizing Key Performance Metric Thresholds and Collection.

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 20-34 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 20-34. 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.

Table 20-34 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 20-34 (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 20-34 describes each parameter.

Table 20-35 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 20-34.
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 Editing Thresholds and Collection Options for WebCenter Portal.

Configuring the Frequency of WebLogic Server Health Checks

Out-of-the-box, the general health of the WebLogic Server on which WebCenter Portal 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 20-36 describes each parameter.

Table 20-36 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 Editing Thresholds and Collection Options for WebCenter Portal.

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, 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 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 WebCenter Portal is deployed. Select **Environment** then, **Servers**, and then select the WebCenter Portal instance (`WC_Portal`).
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.

Editing Thresholds and Collection Options for WebCenter Portal

To change metric thresholds and collection criteria for WebCenter Portal:

1. Copy the XML snippet in Example 20-1 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 20-35

- Table 20-36

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**, then **Servers**, and then `WC_Portal`.
   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
      ```

      > **Note:**
      >
      > 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_properties.xml
      > ```

   e. Restart the managed server.

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 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 Using Key Performance Metric Data to Analyze and Diagnose System Health .

Some common performance issues and actions are described in this chapter:

- Understanding Some Common Performance Issues and Actions
- Troubleshooting Common Issues with Tools and Services

For more detailed troubleshooting tips relating to performance, see Troubleshooting WebCenter Portal.

Tuning Oracle WebCenter Portal Performance

See Oracle WebCenter Portal Performance Tuning in *Tuning Performance* for information on tuning WebCenter Portal. 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.

Monitoring Performance Using WebCenter Portal Performance Pack

WebCenter Portal Performance Pack is a performance diagnostics tool that can be integrated seamlessly into the development phase to get the most out of your deployment. It is available as an add-on as part of Oracle WebCenter Portal. Using WebCenter Portal Performance Pack you can quickly identify and address critical performance bottlenecks in your application. For information, see About WebCenter Portal Performance Pack in *Using Oracle WebCenter Portal Performance Pack*.

Improving Data Caching Performance

To enhance performance and scalability, WebCenter Portal uses Coherence by default for its data caching solution. However, the Oracle Coherence license included in WebCenter Portal is *restricted*, which means that by default a Local caching scheme without any distributed data caching is supported. In a High-Availability (HA) environment deployment, the cached entries are not shared across JVMs/machines.

You can however, use the *distributed mode* for better performance in a clustered environment if you have Coherence or WebLogic Suite licensing. This section guides you on how to set up distributed cache and override WebCenter Portal's default caching configuration to improve performance, provided you have the appropriate license.

This section contains the following topics:

- Summary of Coherence Cache Types
- Default Coherence Caches in WebCenter Portal
- Overriding the Default Configuration

> **Note:**
>
> For more information about configuring coherence, see Configuring and Managing Coherence Clusters in *Administering Clusters for Oracle WebLogic Server*.

Summary of Coherence Cache Types

The basic types of cache modes provided by Coherence are outlined in Table 20-37.

Table 20-37 Basic Cache Types

Cache Name	Description
Distributed	Data is partitioned among all the machines of the cluster. For fault-tolerance, partitioned caches can be configured to keep each piece of data on one or more unique machines within a cluster. Distributed caches are the most commonly used caches in Coherence.
Replicated	Data is fully replicated to every member in the cluster. This cache offers the fastest "read" performance with linear performance scalability for "reads," but poor scalability for "writes" (because "writes" must be processed by every member in the cluster). Because data is replicated to all machines, adding servers does not increase aggregate cache capacity.
Optimistic	Similar to the replicated cache, but without any concurrency control. This implementation offers higher write throughput than a replicated cache. It also allows using an alternative underlying store for the cached data (for example, a MRU/MFU-based cache). However, if two cluster members are independently pruning or purging the underlying local stores, it is possible that a cluster member may have different store content than that held by another cluster member.
Near	A near cache is a hybrid cache; typically fronts a distributed cache or a remote cache with a local cache. Near cache backed by a partitioned cache offers zero-millisecond local access for repeat data access, while enabling concurrency and ensuring coherency and fail-over, effectively combining the best attributes of replicated and partitioned caches.
Local	A local cache is a cache that is local to (completely contained within) a particular cluster node. While it is not a clustered service, the Coherence local cache implementation is often used in combination with various clustered cache services.

For more information about the types of caches provided by Coherence, see Introduction to Coherence Caches in *Developing Applications with Oracle Coherence* guide.

Default Coherence Caches in WebCenter Portal

The default user-configurable Coherence cache entries for WebCenter Portal are shown in Table 20-38.

Table 20-38 Default Coherence Caches in WebCenter Portal

Cache Name	Purpose	Default Coherence Configuration
`oracle.webcenter.spaces.model.ApplicationSpaceObjects`	Cache for Application Space	`WebCenter_12HourCache`
`oracle.webcenter.spaces.model.SpaceProperties`	Cache for Space Properties	`WebCenter_12HourCache`
`oracle.webcenter.genericsiteresources`	Cache for Generic Site Resources	`WebCenter_12HourCache`
`oracle.webcenter.profile`	Cache for People Profile	`WebCenter_12HourCache`

Table 20-38 (Cont.) Default Coherence Caches in WebCenter Portal

Cache Name	Purpose	Default Coherence Configuration
`oracle.webcenter.doclib.provisioned`	Doc lib caches (Provisioned and configured)	`WebCenter_12HourCache`
`oracle.webcenter.page`	Cache for Page definitions	`WebCenter_12HourCache`

The properties of the default Coherence configuration shown in Table 20-38 are described as follows:

Default Configuration	Eviction Policy	High Units	Expiration Delay
WebCenter_12HourCache	Hybrid	1000	12 hours
WebCenter_60MinuteCache	Hybrid	1000	1 hour

Where:

- High Units is the maximum number of units that can be placed in the cache before pruning occurs
- Hybrid Eviction Policy chooses which entries to evict based on the combination (weighted score) of how often and how recently they were accessed. Those entries that are accessed least frequently and those that were not accessed for the longest period are evicted first.
- Expiration Delay specifies the amount of time from the last update that entries will be kept by the cache before being marked as expired. Any attempt to read an expired entry will result in a reloading of the entry from the configured cache store. Expired values are periodically discarded from the cache.

Coherence can be deployed with a standalone application, as an application server library or part of a Java EE module within an `EAR` or `WAR` file or also within the WebLogic Server context.

Overriding the Default Configuration

By default, WebCenter Portal uses the local data caching mode. To use the *distributed mode* for better performance in a clustered environment, you can override the default configuration.

To override the default configuration:

1. Configure WebLogic Clusters (as needed by High Availability configuration) and Coherence clusters.

   For more information, see Configuring and Managing Coherence Clusters in *Administering Clusters for Oracle WebLogic Server*.

2. Define a cache configuration file to override the default configuration.

   For more information, see Configuring Caches and Cache Configuration by Example in *Developing Applications with Oracle Coherence*.

   By default, local caching is used. Make sure to use the exact names of the available caches provided in Table 20-38 in your cache configuration file.

   For example, the people profile cache is uniquely identified by the name `oracle.webcenter.profile` and maps to the local scheme `WebCenter_60MinuteCache` by

default. If you have a large number of users working on the portal, you might want to cache the users' profiles as recreating this object is expensive. By increasing the size for this cache, you can retain the user profile objects in the cache and achieve better performance in the Activity Stream.

To override this cache to use a distributed scheme, make sure to specify the cache name correctly.

```
<cache-mapping>
     <cache-name>oracle.webcenter.profile</cache-name>
     <scheme-name>my_distributed_scheme</scheme-name>
</cache-mapping>
```

where, `my_distributed_scheme` refers to your newly defined distributed caching scheme.

3. After the configuration file is defined, override the default cache configuration file by placing it at `[APPLICATIONS_HOME]/wc_domain/custom.webcenter.spaces.fwk/APP-INF/classes`.

> **Note:**
>
> Both WebCenter Portal and WebCenter Content Presenter can use the same coherence cache configuration file.

For more information, see Create cluster cache configurations in *WebLogic Server Administration Console Online Help*.

Any configuration changes persist and will remain even after an upgrade of WebCenter Portal.

21

Managing WebCenter Portal Logs

Configure diagnostic logging and error messages in WebCenter Portal.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

> **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 Understanding Administrative Operations, Roles, and Tools.

Topics:

- Introduction to Diagnostic Logging
- Viewing and Configuring Log Information

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 Managing Log Files and Diagnostic Dat in *Administering Oracle Fusion Middleware*.

See also, Understanding the Diagnostic Frameworka in *Administering Oracle Fusion Middleware*.

This section includes the following topics:

- WebCenter Portal Diagnostics Log
- Oracle WebCenter Portal Message IDs
- Out-Of-Bound Conditions for Oracle WebCenter Portal Performance Metrics

WebCenter Portal Diagnostics Log

The diagnostics log file for WebCenter Portal is `WC_Portal-diagnostic.log`.

This log is available under the *`DOMAIN_HOME`*`/servers/WC_Portal/logs` directory.

Oracle WebCenter Portal Message IDs

Oracle WebCenter Portal log messages fall into these categories:

Table 21-1 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

Table 21-1 (Cont.) Oracle WebCenter Portal Message Categories

Message ID Range	Message Category
Wiki	WCS-30001 ~ WCS-31000
WebCenter Portal Editor	WCS-31001 ~ WCS-32000
Worklist	WCS-32001 ~ WCS-33000
Content Adapters	WCS-34001 ~ WCS-35000
VCR ADF Integration	WCS-35001 ~ WCS-36000
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
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
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_Portal] [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_Portal] [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_Portal] [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_Portal] [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 myPortlets is out-of-bounds
```

Viewing and Configuring Log Information

This section includes the following topics:

- Viewing and Configuring WebCenter Portal Logs
- Viewing and Configuring Error Messages in WebCenter Portal

Viewing and Configuring WebCenter Portal Logs

To view log messages for a WebCenter Portal application:

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

   See 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, navigate to the home page for WebCenter Portal.

   See 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 Viewing and Searching Log Files in *Administering Oracle Fusion Middleware*.

Viewing and Configuring Error Messages in WebCenter Portal

To help developers debug WebCenter Portal assets, administrators can enable error messages with the calling stack to be displayed in the WebCenter Portal error page.

> **Caution:**
>
> For security reasons, error messages should not be enabled in a production environment. Oracle recommends that you restrict error messages to development and staging environments.

To enable error messages:

1. Connect to the WebCenter domain's Administration server using WLST.
2. Create a new folder (`/tmp/WCconfig`) on your local file system.
3. Export the `webcenter-config.xml` configuration file to the `/tmp/WCconfig` folder you created by running:

   ```
   exportMetadata(application='webcenter', server='WC_Portal', toLocation='/tmp/
   WCconfig', docs='/oracle/webcenter/webcenterapp/metadata/webcenter-config.xml')
   ```

4. Open the `webcenter-config.xml` file (in the `/tmp/WCconfig/oracle/webcenter/webcenterapp/metadata`) folder and change the `showError-enable` property to `true` to enable error messages:

   ```
   <webcenter:showError-enabled>true</webcenter:showError-enabled>
   ```

   or `false` to disable it:

   ```
   <webcenter:showError-enabled>false</webcenter:showError-enabled>
   ```

5. Save the file and import it back to the Administration server using the following WLST command:

   ```
   importMetadata(application='webcenter', server='WC_Portal',fromLocation='/tmp/
   WCconfig', docs='/oracle/webcenter/webcenterapp/metadata/webcenter-config.xml')
   ```

   Note that you do not need to restart the Administration server for the change to take effect.

22

Managing WebCenter Portal Audit Logs

Configure, manage, and interpret audit logging for WebCenter Portal.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

> **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 Understanding Administrative Operations, Roles, and Tools.

Topics:

- Introduction to Managing Audit Logs
- Configuring Audit Logging
- Viewing WebCenter Portal Audit Events

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 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 *Securing Applications with Oracle Platform Security Services*. For information about configuring the Audit Service to use a database, see Configuring and Managing Auditing in *Securing Applications with Oracle Platform Security Services*.

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 includes the following topics:

- Setting the Logging Level
- Configuring the Audit Store Database

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 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",filterPreset="Low")
```

Set the logging level to `Medium`:

```
setAuditPolicy(componentType="webcenter",filterPreset="Medium")
```

To turn logging off for WebCenter Portal:

```
setAuditPolicy(componentType="webcenter",filterPreset="None")
```

Successful execution does not throw any error and completes silently. Restart the `WC_Portal` 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 *Securing Applications with Oracle Platform Security Services*.

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_Portal/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 *Securing Applications with Oracle Platform Security Services*.

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.

Viewing WebCenter Portal Audit Events

This section describes the WebCenter Portal audit 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:

- Using WebCenter Portal Audit Logs
- Querying the Audit Schema

Using WebCenter Portal Audit Logs

Table 22-1 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, wikis and blogs, 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 are logged:

- PortalLifeCycle
- PortalRoleManagement
- PortalRoleMemberManagement
- PortalToolAccessManagement
- ImpersonationSessionMgmt

When the log level is set to `Medium`, events in the following additional categories are logged:

- PortalToolsManagement
- PortalPagesManagement

Table 22-1 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

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")
```

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_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_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_AUDITVIEW where EventType = 'DeletePortal';
     ```

   - The following SQL statement returns all audit data for WebCenter Portal:

     ```
     select * from webcenter_AUDITVIEW;
     ```

If you want to regularly monitor WebCenter Portal activities you can create a SQL data source using SQL queries and drop the data source as a table or other visualization onto a portal page. For more information about SQL data sources, see Working with Data Sources in *Building Portals with Oracle WebCenter Portal*.

Part V
Administering Security

This part of *Administering Oracle WebCenter Portal* provides information about the security administration topics for Oracle WebCenter Portal:

- Managing WebCenter Portal Security
- Configuring the Identity Store
- Managing Users and Application Roles
- Configuring Single Sign-On
- Configuring SSL
- Configuring Web Services Security
- Configuring Security for Portlet Producers
- Managing Impersonation

23
Managing WebCenter Portal Security

Understand how WebCenter Portal is secured and learn how to configure and manage security.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

> **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, Understanding Administrative Operations, Roles, and Tools.

Topics:

- Introduction to Application Security
- Default Security Configuration

For information about specific aspects of configuring security for WebCenter Portal, see:

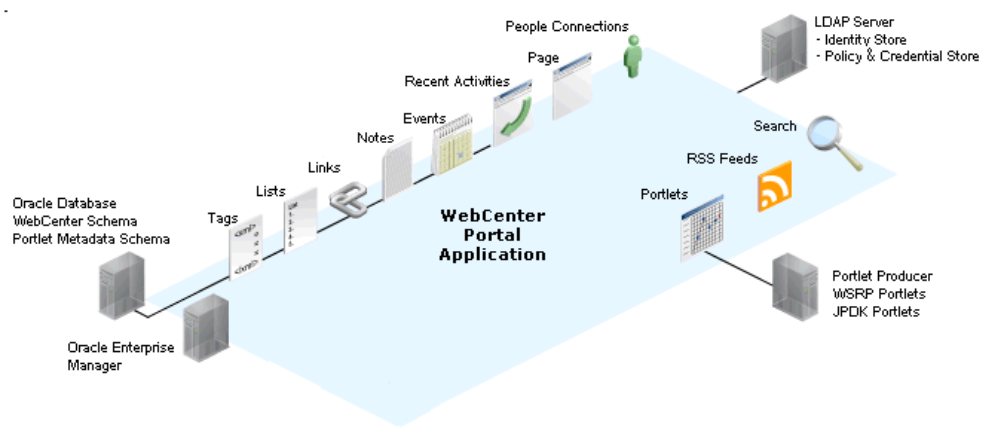
- Configuring the Identity Store
- Managing Users and Application Roles
- Configuring Single Sign-On
- Configuring SSL
- Configuring Web Services Security
- Configuring Security for Portlet Producers

Introduction to Application Security

The recommended security model for WebCenter Portal is based on Oracle ADF Security, which implements the Java Authentication and Authorization Service (JAAS) model. For more information about Oracle ADF Security, see Introduction to Oracle ADF in *Developing Fusion Web Applications with Oracle Application Development Framework*.

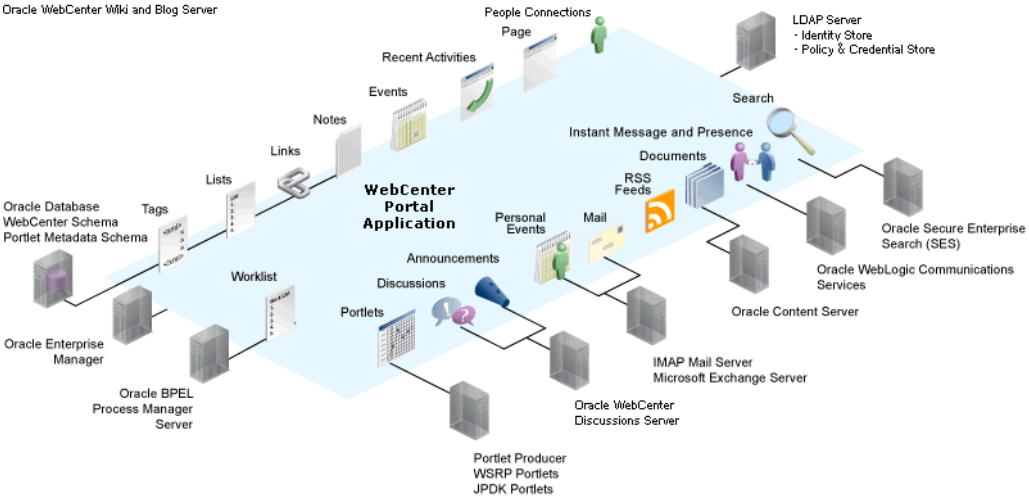
Figure 23-1 shows the relationship between a WebCenter Portal application deployment and its services, servers, portlets, portlet producers, its identity, credential and policy stores, and Oracle Enterprise Manager.

Figure 23-1 Basic WebCenter Portal Application Architecture



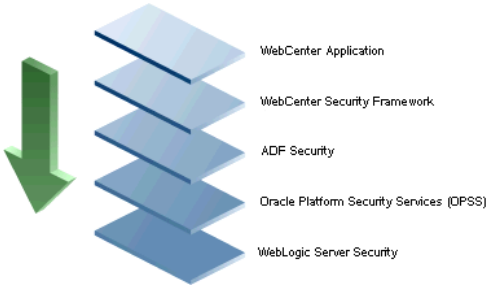
The diagram in Figure 23-2 shows a basic WebCenter Portal application after deployment with its back-end server connections.

Figure 23-2 WebCenter Portal Application Architecture with Back-End Server Connections



The diagram in Figure 23-3 shows the security layers for a WebCenter Portal application.

Figure 23-3 WebCenter Portal Security Layers



WebCenter Portal 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

Default Security Configuration

This section describes the security configuration that is in place when a WebCenter Portal application is deployed, and the configuration tasks that should be carried out after deployment:

- Administrator Accounts
- Application Roles and Enterprise Roles
- Default Identity and Policy Stores
- Default Policy Store Permissions and Grants
- Post-deployment Security Configuration Tasks

Administrator 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 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.

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 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.

Default Identity and Policy Stores

By default, WebCenter Portal is configured to use a file-based embedded LDAP identity store to store application-level user IDs, and an Oracle RDBMS (releases 10.2.0.4 or later; releases 11.1.0.7 or later; and releases 11.2.0.1 or later) 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 list of supported versions of identity store types, see Oracle Fusion Middleware 12c Certifications.

> **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 Configuring the Identity Store.

The policy and credential stores can use either the default database store or Oracle Internet Directory 11gR1 or 10.1.4.3. 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 Supported LDAP-, DB-, and File-Based Services in *Securing Applications with Oracle Platform Security Services*. For more information on reconfiguring the identity store and the policy and credential stores, see Configuring the Identity Store and Managing Users and Application Roles.

> **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 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 Migrating the Discussions Server to Use an External LDAP.
>
> Both WebCenter Portal and Content Server must share the same LDAP server. For more information, see Configuring Oracle WebCenter Content to Share the WebCenter Portal Identity Store LDAP Server.

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 topics:

- Permission-based Authorization
- Role-mapping Based Authorization
- Default Policy Store Permissions for WebCenter Portal
- Default Code-based Grants

Permission-based Authorization

Permission-based authorization is used for tools, such as lists, where access control is implemented within the WebCenter Portal 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 Managing Security Across Portals.

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 WebCenter Portal users (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.

For more information, see Understanding Discussion Server Role Mapping.

Default Policy Store Permissions for WebCenter Portal

Out-of-the box, WebCenter Portal provides the following default roles:

Default application roles:

- Administrator
- Application Specialist
- Portal Creator
- Authenticated-User
- Public-User

For more information about the default application roles, see Managing Security Across Portals.

Default role in a portal:

- Portal Manager

> **Note:**
>
> The portal-level roles of `Participant` and `Viewer` are no longer created by default. In order to create portals faster and eliminate unneeded roles, there are fewer default portal-level roles created by default.

Default Code-based Grants

WebCenter Portal makes 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).

Similarly, WebCenter Portal 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.

Post-deployment Security Configuration Tasks

After deploying WebCenter Portal, 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 uses an embedded LDAP for the 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 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 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 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 Configuring Oracle WebCenter Content to Share the WebCenter Portal Identity Store LDAP Server.

- **Configuring SSO**

  Single Sign-On (SSO) lets users log in once across WebCenter Portal and components rather than having to log in for each sub-application (for example, to accessing a wiki page). 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 supports four single sign-on solutions: Oracle Access Manager (OAM), Oracle Single Sign-on (OSSO), a SAML-based single sign-on solution, 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 Configuring Single Sign-On.

- **Configuring SSL**

  Secure Sockets Layer (SSL) provides additional security for connections between WebCenter Portal and 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 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.

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 Oracle WebCenter Portal Performance Tuning in *Tuning Performance*.

This section includes the following topics:

- Setting the Policy Store Refresh Interval
- Setting the Connection Pool Cache
- Setting User Cache Settings
- Setting Group Cache Settings

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 Refreshing the Policy Cache in *Securing Applications with Oracle Platform Security Services*.

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.

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.

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.

24

Configuring the Identity Store

Associate the identity store with an external LDAP instead of the default embedded LDAP identity store, configure an LDAP server for Oracle WebCenter Content Server, and use Oracle Identity Cloud Service as the identity store for WebCenter Portal.

Note:

Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

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.

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 Understanding Administrative Operations, Roles, and Tools.

Topics:

- Reassociating the Identity Store with an External LDAP Server
- Configuring the GUID Attribute for External LDAP Identity Stores
- Adding Users to the Embedded LDAP Identity Store
- Moving the Administrator Account to an External LDAP Server

- Configuring Oracle WebCenter Content to Share the WebCenter Portal Identity Store LDAP Server
- Aggregating Multiple Identity Store LDAP Servers Using libOVD
- Configuring Dynamic Groups for WebCenter Portal
- Configuring the REST Service Identity Asserter

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, 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_Portal` server, Content Server, and Collaboration server must all be configured to use the same external LDAP server.

It is recommended that you set a strong password policy on the LDAP server for the identity store. Oracle recommends that user passwords meet the following requirements:

- Passwords should not contain the user's account name or parts of the user's full name that exceed two consecutive characters.
- Passwords must be at least six characters in length or the number of characters specified in the minimum password length policy setting.
- Enforce password history policy setting, which determines the number of unique new passwords that have to be associated with a user account before an old password can be reused. The setting for this value can be between 0–24 (if this value is set to 0, Enforce password history is disabled; a higher value, such as 24, is preferable to prevent security vulnerability through password reuse).
- Passwords must contain characters from at least three of the following four categories: English uppercase alphabet characters (A to Z), English lowercase alphabet characters (a to z), base 10 digits (0 to 9), non-alphanumeric characters (for example, !$#,%) .

For the GUID attribute for other supported LDAPs, see Configuring the GUID Attribute for External LDAP Identity Stores. For other user attribute mappings for supported LDAP servers, see the User and Role API Reference in *Securing Applications with Oracle Platform Security Services*.

> **Note:**
>
> To use an existing database (i.e., not a default database store created when WebCenter Portal is installed in its default configuration) for the identity store, you must either use OVD or write a custom provider based on the User and Role API. Note that LibOVD should not be used in conjunction with a database identity store.

> **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 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`.

> **Note:**
>
> If using iPlanet, set the `virtualize` property to `true` in `./user_projects/domains/soainfra/config/fmwconfig/jps-config.xml`.
>
> ```
> <serviceInstance name="idstore.ldap"
> provider="idstore.ldap.provider">
>  <property name="idstore.config.provider"
> value="oracle.security.jps.wls.internal.idstore.WlsLdapIdStoreConfig
> Provider"/>
>  <property name="CONNECTION_POOL_CLASS"
> value="oracle.security.idm.providers.stdldap.JNDIPool"/>
>  <property name="virtualize" value="true"/>
>  <property name="OPTIMIZE_SEARCH" value="true"/>
> </serviceInstance>
> ```

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.
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 OPSS System and Configuration Properties appendix in *Securing Applications with Oracle Platform Security Services*.

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)
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 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.

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 User and Role API Reference section in *Securing Applications with Oracle Platform Security Services*. See also Mapping User Attributes to LDAP Directories in *Securing Applications with Oracle Platform Security Services*. Note that as shown in the table in Mapping User Attributes to LDAP Directories, not all attributes are available across all LDAP servers, including the embedded LDAP server that comes with WebLogic Server (WLS).

> **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`

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 *Securing Applications with Oracle Platform Security Services*.

For Oracle Internet Directory, users are typically managed using ODSM (described in Managing Directory Entries in *Administering Oracle Internet Directory*).

> **Note:**
>
> If you are planning to reassociate your identity store with an external LDAP, perform that step first (as described in 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 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:

- Adding Users to the Identity Store Using the WLS Administration Console
- Adding Users to the Identity Store Using an LDIF File

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 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.

12. 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. Find the `serviceInstance` entry for `idstore.ldap.provider` and add the following two properties.

    ```
    <property name="username.attr" value="sAMAccountName"/>
    <property name="user.login.attr" value="sAMAccountName"/>
    ```

    The following is a sample of how your entries would look like:

    ```
    <serviceInstances>
     <!-- JPS WLS LDAP Identity Store Service Instance -->
     <serviceInstance name="idstore.ldap" provider="idstore.ldap.provider">
     <property name="idstore.config.provider"
    ```

```
value="oracle.security.jps.wls.internal.idstore.WlsLdapIdStoreConfigProvider"/
>
 <property name="username.attr" value="sAMAccountName"/>
 <property name="user.login.attr" value="sAMAccountName"/>
</serviceInstance>
```

c. Save `jps-config.xml`.

d. Restart the Admin Server and the managed servers.

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.
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

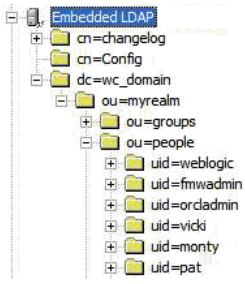
Create an LDIF File

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 24-1.

Figure 24-1 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 Oracle Internet Directory Data Management Tools in *Reference for Oracle Identity Management*. For a complete list of user attribute mappings for LDAP servers supported by WebCenter Portal, see Mapping User Attributes to LDAP Services in the *Securing Applications with Oracle Platform Security Services*.

```
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
```

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 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 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 Changing the Administrator Group Name.

  > **Note:**
  >
  > Since OWSM is dependent on the OracleSystemUser and OracleSystemGroup entities, which are provided by the DefaultAuthenticator, to get OWSM working after the embedded LDAP is removed you'll need to modify the default user. For more information, see Modifying the Default User in *Securing Web Services and Managing Policies with Oracle Web Services Manager*.

This section includes the following topics:

- Migrating the Discussions Server to Use an External LDAP
- Changing the Administrator Group Name

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 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 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:

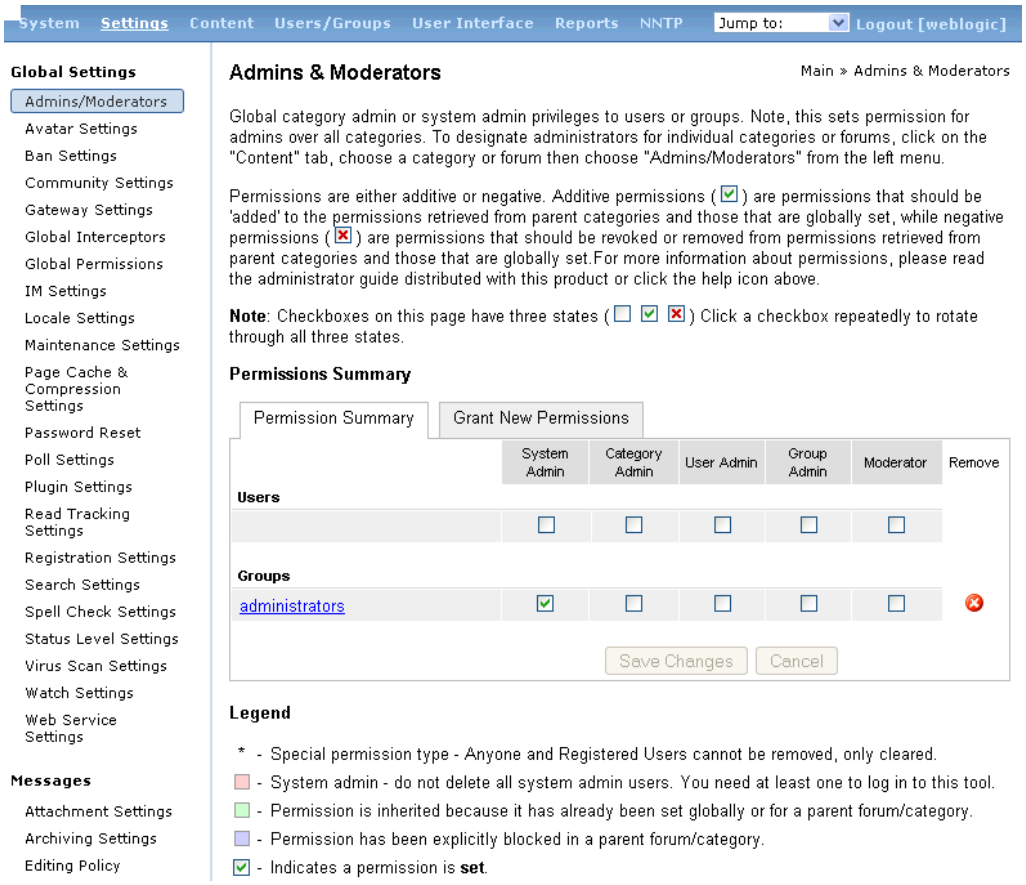
   `http://host:port/owc_discussions/admin`

   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 24-2).

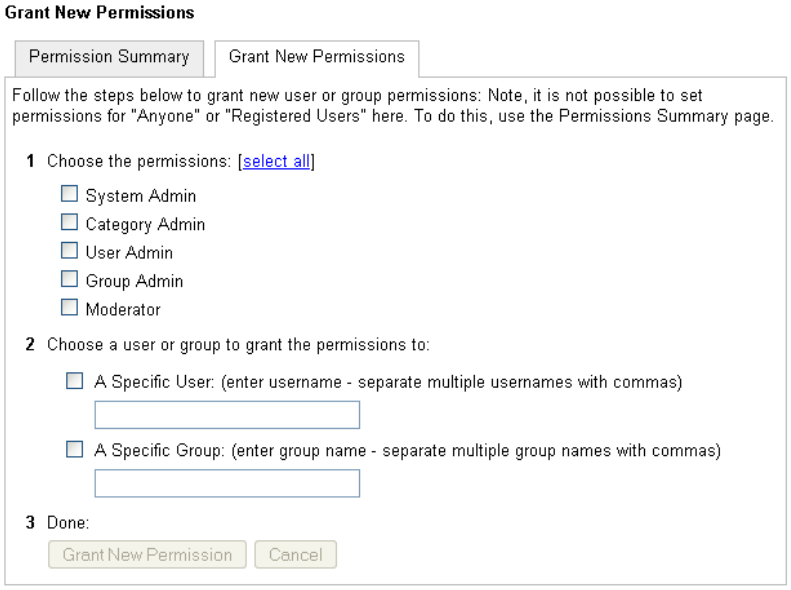
Figure 24-2 Admins & Moderators Page



5. Click **Grant New Permissions**.

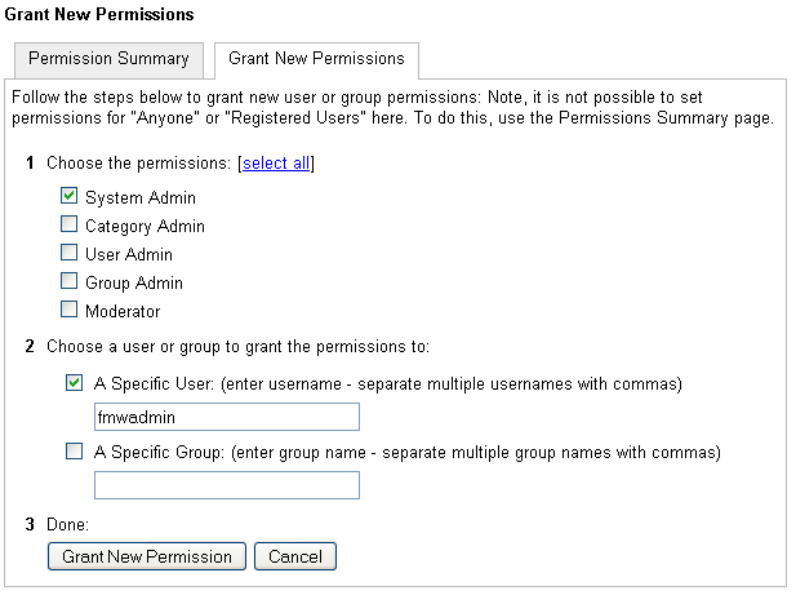
   The Grant New Permissions pane displays (see Figure 24-3).

Figure 24-3 Grant New Permissions Pane



6. Grant System Admin privileges to the user you created, as shown in Figure 24-4.

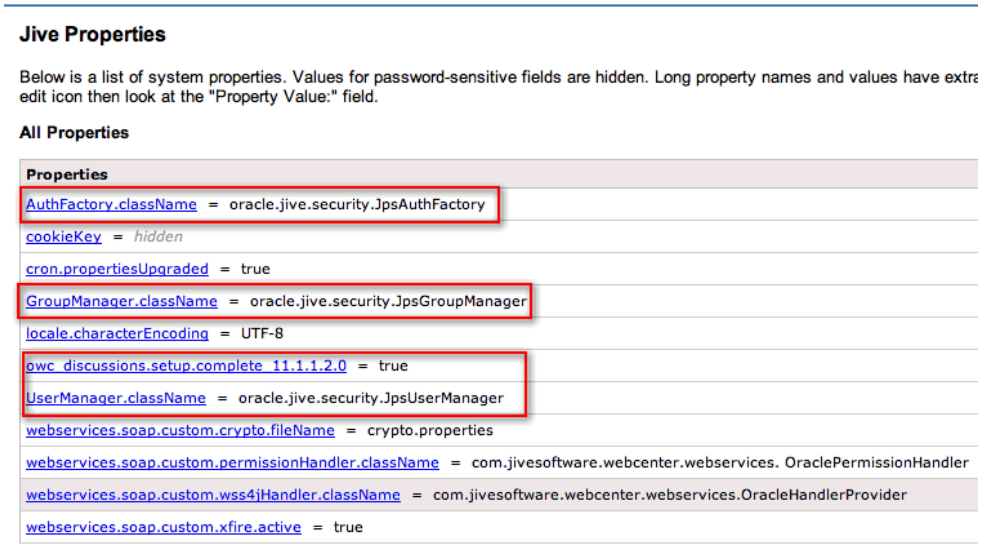
Figure 24-4 Grant New Permissions Pane with New User



7. Click **System > System Properties**.

   The Jive Properties page displays (see Figure 24-5).

Figure 24-5 Jive Properties Page



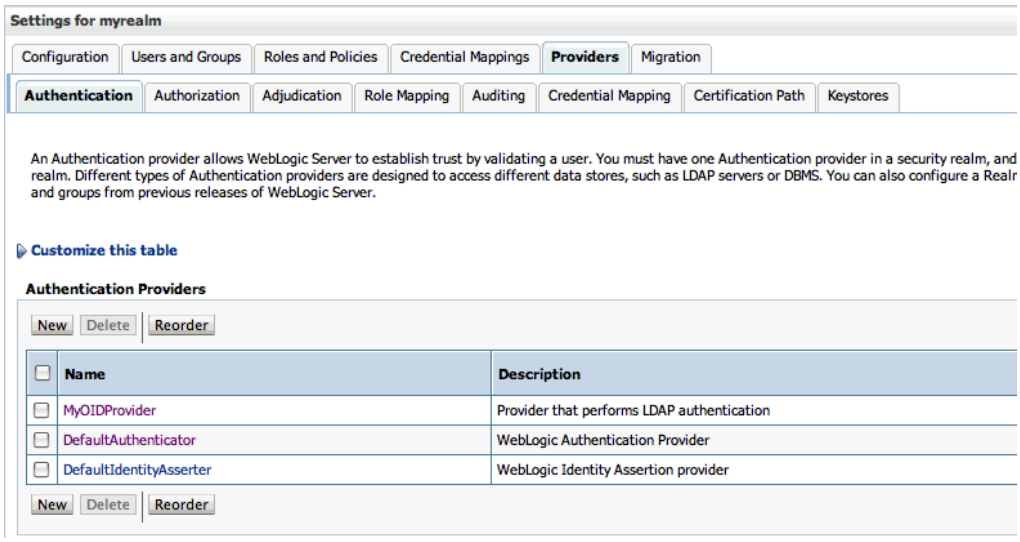
8. Check that the properties marked in red have been added and are set as shown in Figure 24-5.
9. Log in to the WebLogic Server Administration Console.

   For information on logging in to the WebLogic Server Administration Console, see 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** subtab, 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 24-6:

Figure 24-6 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 Granting the WebCenter Portal Administrator Role).

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 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.

Configuring Oracle WebCenter Content to Share the WebCenter Portal Identity Store LDAP Server

The WebCenter Content server must be configured to use the same identity store LDAP server as WebCenter Portal. For more information on configuring WebCenter Content, see Managing Connections to Oracle WebCenter Content Server and also see Configuring the LDAP Identity Store Service in *Securing Applications with Oracle Platform Security Services*.

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 Users Cannot Self-Register when WebCenter Portal Configured with Active Directory.

This section contains the following topics:

- Configuring libOVD for Identity Stores with Complete User Profiles
- Configuring libOVD for Identity Stores with Partial User Profiles
- Restoring the Single Authenticator

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."

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.

Restoring the Single Authenticator

You can restore the single authenticator by removing the Join Adapter rule, thereby backing out the configuration done in 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.

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 topics:

- Creating a Dynamic Group Using an LDIF File
- Creating a Dynamic Group Using the Oracle Directory Services Manager

> **Note:**
>
> Dynamic groups is not supported for LDAPs other than OID unless OVD is used.

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: Defining a Dynamic Group Using an LDIF File

```
dn:
cn=managers,cn=portal.070720.104824.056918000,cn=groups,dc=us,dc=oracle,dc=co
m
labeleduri: 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 the*Administering 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: Updating OID Using the ldapadd Command

```
ldapadd -h myserver -p 12061 -D cn=fmwadmin -w mybiz1 -f managers.ldif -v
add labeleduri: ldap://myserver.example.com:12061/cn=users,dc=us,dc=mybiz,dc=com??
sub?(title=Manager)
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
```

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 Using Oracle Directory Services Manager in *Administering 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.

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 applications requires that an identity asserter be configured for it in the WebCenter domain identity store. The following topics show how to configure OPSS Trust Service instances and identity asserters for Oracle WebLogic Server.

This section contains the following topics:

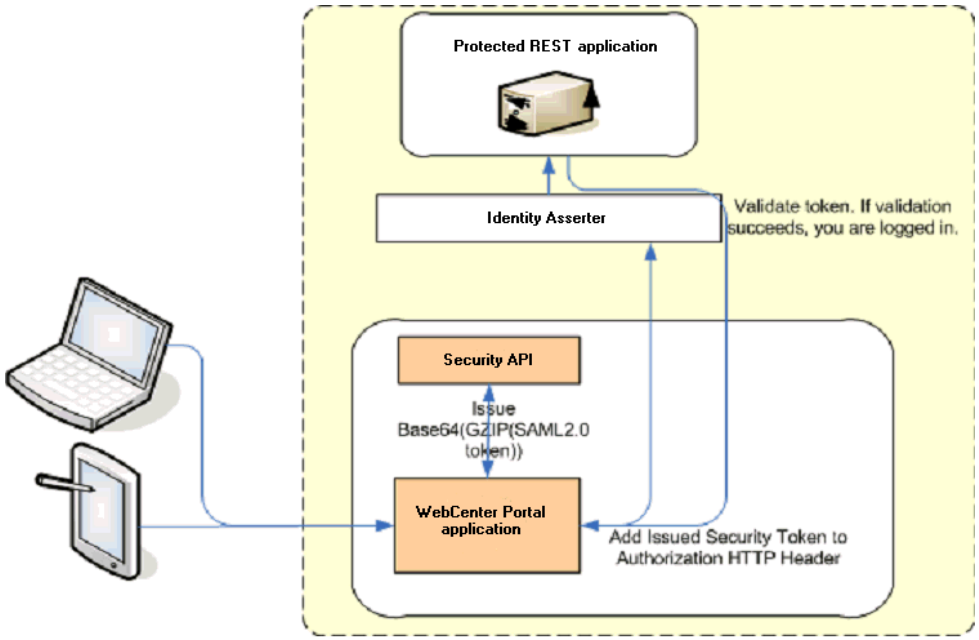
- Understanding the REST Service Instance and Identity Asserter
- Setting up the Client Application
- Configuring the WLS Trust Service Asserter

Understanding the REST Service Instance and Identity Asserter

Although WebCenter Portal 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 24-7 shows the different components involved in the identity propagation and assertion.

Figure 24-7 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 applications) connect to a WebCenter Portal 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 uses the response and shows the page to the end user.

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 as shown in Creating the WebCenter Portal Domain Keystore, 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..

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.

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.

25

Managing Users and Application Roles

You can grant WebCenter Portal Administrator role to a user and add users to application roles.
In WebCenter Portal, by default the policy and credential store is configured to use a database. A file-based or an LDAP-based policy store is no longer supported. When migrating from a WebCenter Portal 11*g* release if your 11*g* instance is configured to use a file-based or an LDAP-based policy store, you must migrate to a database policy store. A database policy store is supported for both single and high availability (HA) environments.

For information about configuring an Oracle database as the policy and credential store, see Using a Database-Based Security Store and Reassociating the Security Store in *Securing Applications with Oracle Platform Security Services*. For troubleshooting information, see Reassociation Failure in *Securing Applications with Oracle Platform Security Services*. Migrating a file-based or an LDAP-based policy store to a database-based policy store is a pre-upgrade task as described in Performing the Oracle WebCenter Pre-Upgrade Tasks in *Upgrading Oracle WebCenter*.

> **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 Understanding Administrative Operations, Roles, and Tools.

Topics:

- Granting the WebCenter Portal Administrator Role
- Granting Application Roles
- Using the Runtime Administration Pages
- Configuring Self-Registration By Invitation in WebCenter Portal

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:

- Granting the WebCenter Portal Administrator Role Using Fusion Middleware Control

- Granting the WebCenter Portal Administrator Role Using WLST

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:

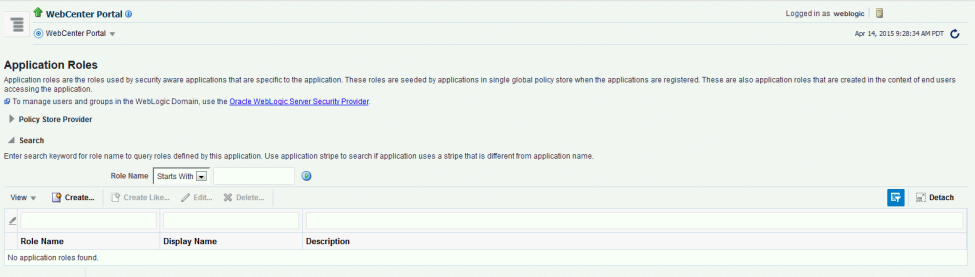
1. Log into Fusion Middleware Control and navigate to the WebCenter Portal home page.

   See Navigating to the Home Page for WebCenter Portal.

2. From the WebCenter Portal menu, select **Security** and then **Application Roles**.

   The Application Roles page opens (see Figure 25-1).

Figure 25-1 Application Roles Page



3. Search for the WebCenter Portal Administrator role:

   - In the **Role Name** field, enter the following internal identifier for the Administrator role, and then click the **Search** (arrow) icon:

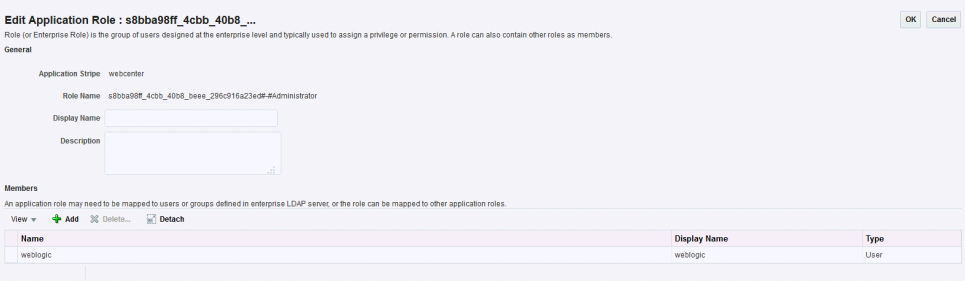
     ```
     s8bba98ff_4cbb_40b8_beee_296c916a23ed#-#Administrator
     ```

   The search should return `s8bba98ff_4cbb_40b8_beee_296c916a23ed#-#Administrator`, which is the administrator role identifier.

4. Click the administrator role identifier from the search results and click **Edit**.

   The Edit Application Role page opens (see Figure 25-2).

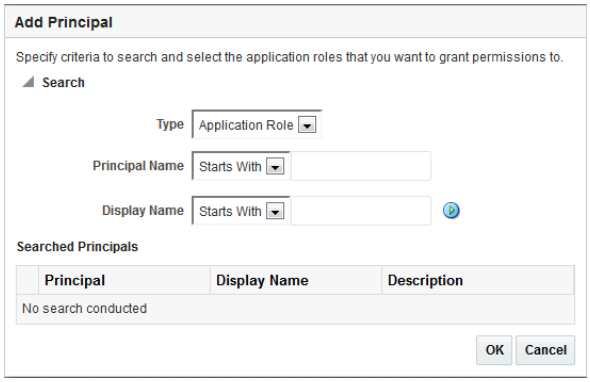
Figure 25-2 Edit Application Role Page



5. Click **Add** from the **Members** section.

   The Add Principal dialog opens (see Figure 25-3).

Figure 25-3 Add Principal Dialog



6. Search for the user to assign the Administrator role to.
   a. From the **Type** drop-down, select **User**.
   b. Enter search criteria in the **Principal Name** and/or **Display Name** fields to either include part of the user name and/or the initial characters of the user name.
   c. Optionally, when you select User, select the **Check to enter principal name here** option from the **Advanced Option** section, enter your search criteria in the **Principal Name** and/or **Display Name** fields.
   d. Click **OK**.

      The Add Principal dialog closes and the user name is added to the list of members.
7. To remove the `weblogic` role from the Edit Application Role page, select the role and click **Delete**, then click **Yes** on the confirmation dialog.
8. On the Edit Application Role page, click **OK**.

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 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")
```

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 topics:

- Granting Application Roles Using Fusion Middleware Control
- Granting Application Roles Using WLST

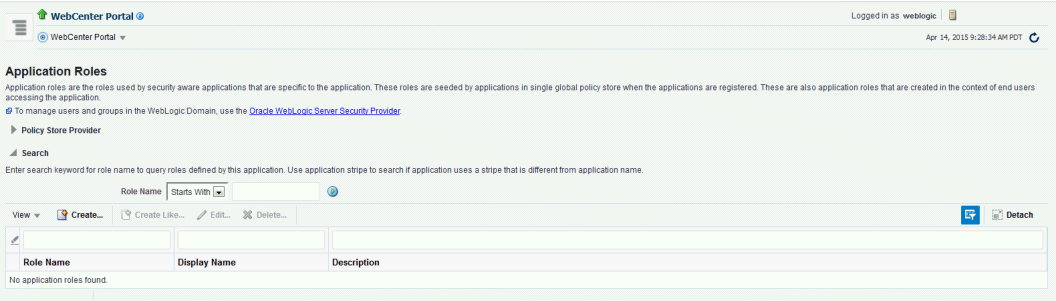
Granting Application Roles Using Fusion Middleware Control

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.
2. From the WebCenter Portal menu, select **Security** and then **Application Roles**.

   The Application Roles page opens.

Figure 25-4 Application Roles Page

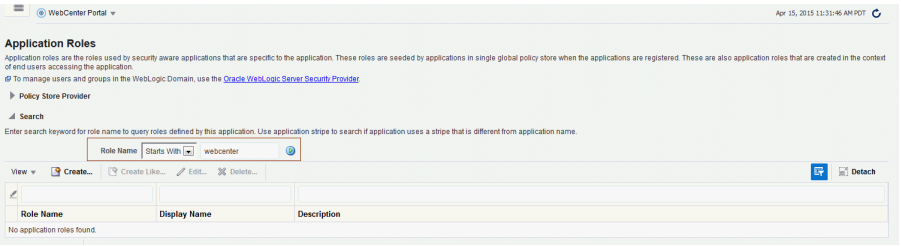


3. In the **Role Name** field, enter `webcenter` to search for all application roles in WebCenter Portal, or enter the name of the role (for example, `appConnectionManager`), and then click the **Search** (arrow) icon: .

   If you are not sure of the name, enter a partial search term or leave the field blank to display all the application roles.

   The Application Roles page opens.

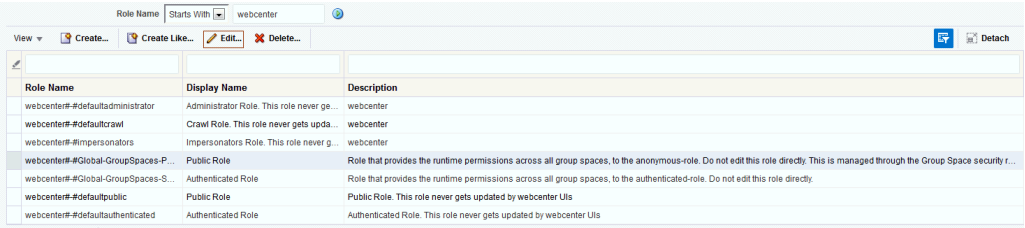
Figure 25-5 Application Roles Page



4. Select the role you want to add the user to, then click **Edit**.

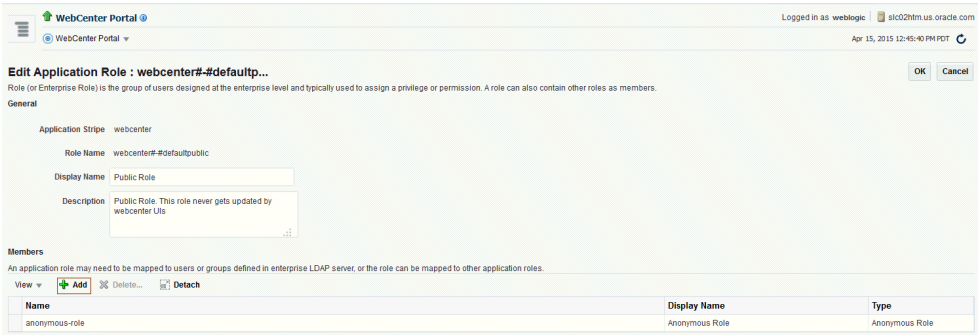
   For example, to add a user to the Public Role, select the row Public Role.

Figure 25-6 Role Name Search Results



5. In the Edit Application page that opens for the selected role, click **Add** .

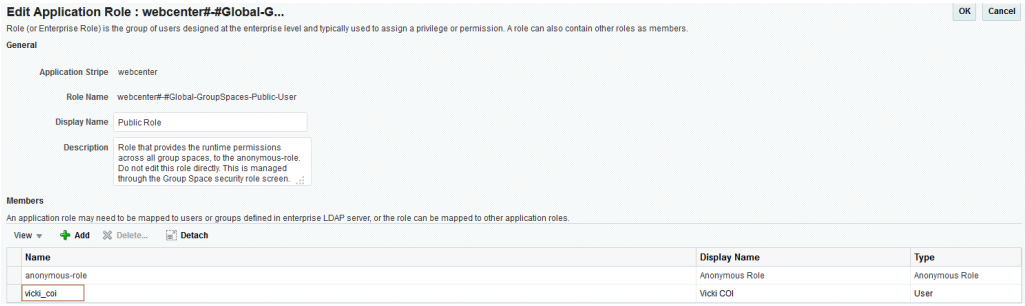
Figure 25-7 Edit Application Role Page



6. In the Add Principal dialog that opens, search for the user.
   a. From the **Type** drop-down, select **User**.
   b. Enter search criteria in the **Principal Name** and/or **Display Name** fields to either include part of the user name and/or the initial characters of the user name.
   c. Select the user name from the Searched Principals table, then click **OK**.

      The Add Principal dialog closes and the user name is added to the list of members for the application role on the Edit Application Role page.

Figure 25-8 User Added to Application Role



7. On the Edit Application Role page, click **OK**.
8. Restart the WebCenter Portal (`WC_Portal`) managed server.

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 *WLST Command Reference for WebLogic Server*.

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. See About WebCenter Portal Security.

> **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 Configuring Profile.

Configuring Self-Registration By Invitation in WebCenter Portal

WebCenter Portal supports self-registration by invitation, as described in Enabling Self-Registration By Invitation-Only. 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 **Allow Self-Registration Through Invitations** in WebCenter Portal Administration, 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 “Managing Credentials with WLST Commands in *Securing Applications with Oracle Platform Security Services*.

26

Configuring Single Sign-On

Configure any of the available single sign-on (SSO) solutions for WebCenter Portal.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

> **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 Understanding Administrative Operations, Roles, and Tools.

Topics:

- Introduction to Single Sign-On
- Configuring Oracle Access Manager
- Configuring SAML-based Single Sign-On
- Configuring SSO for Microsoft Clients
- Configuring SSO with Virtual Hosts

Introduction to Single Sign-On

Single sign-on provides authentication across a topology's components allowing users to log in once, rather than having to log in each time they access a component. Without implementing single sign-on, users must provide credentials each time they access components, such as discussions or Content Server, from WebCenter Portal.

Single sign-on can be implemented for WebCenter Portal 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. OAM (in particular,

OAM 11g) is the recommended single sign-on solution for Oracle WebCenter Portal 12*c* installations.

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 and associated Web tools like discussions, 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 log-ins 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.

Configuring Oracle Access Manager

Oracle Access Manager (OAM) provides flexible and extensible authentication and authorization, and provides audit services. This section describes how to configure WebCenter Portal for OAM single sign-on authentication, including how to configure the WebLogic server side and the WebCenter Portal application as the partner application participating in SSO.

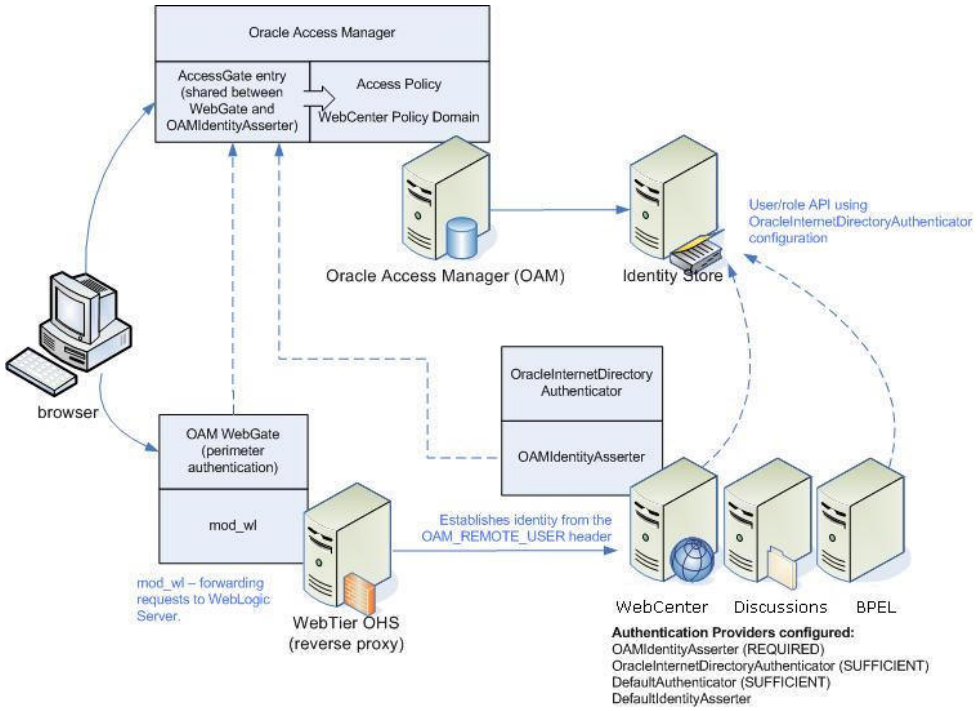
The installation and configuration steps for OAM 11*g* are presented in the following topics:

- OAM Components and Topology
- Roadmap to Configuring OAM
- Installing and Configuring OAM 11g
- Configuring the WebLogic Domain for OAM
- Additional Single Sign-on Configurations
- Testing Your OAM Installation

OAM Components and Topology

Figure 26-1 shows the components and topology required to set up single sign-on with Oracle Access Manager for a WebCenter Portal application.

Figure 26-1 OAM Single Sign-On Components and Topology



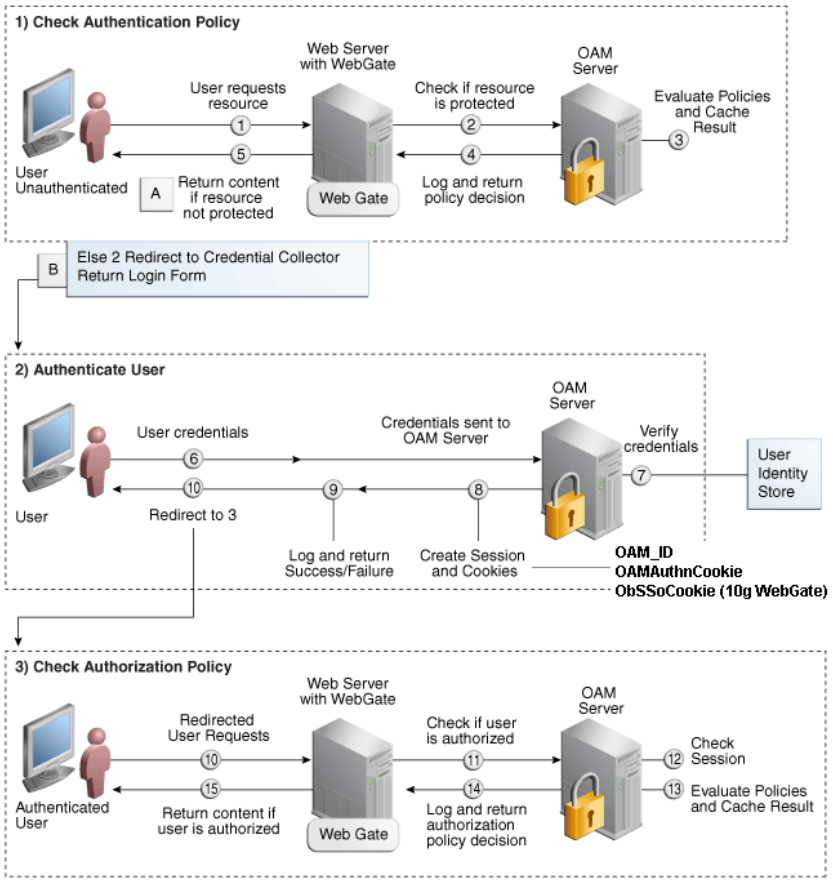
OAM consists of the following components:

- **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 plug-in 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 26-2 shows the single sign-on process flow for OAM.

Figure 26-2 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 application is running, and from where desired content or applications are served to the user, as shown below:

      WebGate -> mod_wl -> WebCenter Portal 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.

Roadmap to Configuring OAM

Table 26-1 provides an overview of the prerequisites and tasks required to configure single sign-on for WebCenter Portal using OAM.

Table 26-1 Configuring Single Sign-on for WebCenter Portal Using OAM

Actor	Task
Administrator	• Installing and Configuring OAM 11g • Installing and Configuring Oracle HTTP Server • Configuring Oracle HTTP Server WebGate • Registering the WebGate Agent
Administrator	Configuring the WebLogic Domain for OAM • Configuring the Oracle Internet Directory Authenticator • Configuring the OAM Identity Asserter • Configuring the Default Authenticator and Provider Order • Adding an OAM Single Sign-on Provider
Administrator	Additional Single Sign-on Configurations
Administrator	Testing Your OAM Installation

Installing and Configuring OAM 11g

This section describes how to install and configure OAM 11g, and includes the following topics:

- Installing and Configuring OAM 11g
- Installing and Configuring Oracle HTTP Server
- Configuring Oracle HTTP Server WebGate
- Registering the WebGate Agent

Installing and Configuring OAM 11g

> **Note:**
>
> OAM should be installed only after you've installed Oracle WebCenter Portal and any other components required for your environment. You should also have configured and tested any required connections.

Install Oracle Access Manager (OAM) as described in Installing and Configuring Oracle Identity Management in *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 Registering a New User Identity Store in *Administering Oracle Access Management*. 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 Installing and Configuring Oracle Identity Management in *Installation Guide for Oracle Identity Management*.

Installing and Configuring Oracle HTTP Server

You can choose to install Oracle HTTP Server 12*c* or Oracle HTTP Server 11*g*. This step should be performed after installing and configuring OAM, and before configuring the WebLogic domain.

To install and configure Oracle HTTP server:

1. Install Oracle HTTP Server 12*c* or Oracle HTTP Server 11*g*.

   To install Oracle HTTP Server 12*c*, see About the Oracle HTTP Server Installation in *Installing and Configuring Oracle HTTP Server*.

To install Oracle HTTP Server 11*g*, see *Installation Guide for Oracle Web Tier*. Oracle HTTP Server is a component of Oracle Web Tier.

2. Update `mod_wl_ohs.conf` to configure web tier OHS so that it forwards requests to the Oracle WebLogic Server for WebCenter Portal. Refer to the example entries given below. Make sure that the WebLogic port numbers match your configuration.

> **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.

```
# 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 /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 /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>

</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.

Configuring Oracle HTTP Server WebGate

You need to configure Oracle HTTP Server WebGate for Oracle Access Manager. Oracle HTTP Server WebGate is a Web server plug-in that intercepts HTTP requests and forwards them to an existing Oracle Access Manager instance for authentication and authorization.

Configuring Oracle HTTP Server 12*c* WebGate

If you installed Oracle HTTP Server 12*c*, the WebGate software is installed as part of the installation. For information about configuring Oracle HTTP Server 12*c* WebGate, see Configuring Oracle HTTP Server WebGate for Oracle Access Manager in *Installing and Configuring Oracle HTTP Server*.

Installing and Configuring Oracle HTTP Server 11*g* WebGate

If you are using Oracle HTTP Server 11*g*, install and configure Oracle HTTP Server 11*g* WebGate for Oracle Access Manager.

1. Install WebGate as described in Installing and Configuring Oracle HTTP Server 11g WebGate for OAM in *Installing WebGates for Oracle Access Manager*. Use the same middleware home that was specified during Oracle HTTP Server installation.

> **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 Registering the WebGate Agent.

2. After installing Oracle HTTP Server 11*g* 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.

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
   ```

Registering the WebGate Agent

After installing the WebGate on the web tier, 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.

> **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.

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 work flows 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 web tier 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 web tier 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 web tier and associated components, continue by configuring the Policy Manager as described in Configuring the WebLogic

Domain for OAM, and performing any additional service and component configurations that apply as described in Additional Single Sign-on Configurations.

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:

- Configuring the Oracle Internet Directory Authenticator
- Configuring the OAM Identity Asserter
- Configuring the Default Authenticator and Provider Order
- Adding an OAM Single Sign-on Provider

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 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))"	The specified user name attribute must match in these three filters: `All Users Filter` and `User Name Attribute` and `User From Name Filter`
User Name Attribute:	"uid"	
User From Name Filter:	"(&(uid=%u) (objectclass=person))"	
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. Additionally, these three fields should also match across all services participating in single sign-on, as well as OAM and WebCenter Portal.

11. Click **Save**.

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 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.

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 Configuring WebCenter Portal for SSO. Also be sure to perform any further service and component configurations that apply to your environment as described in Additional Single Sign-on Configurations.

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.

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.

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 Testing Your OAM Installation.

- Configuring WebCenter Portal for SSO
- Configuring the Discussions Server for SSO
- Configuring SOA Server Connections for SSO
- Configuring OAM for RSS Feeds Using External Readers
- Configuring the WebLogic Server Administration Console and Enterprise Manager for OAM 11g
- Configuring Elasticsearch for SSO
- Configuring Content Server for SSO
- Restricting Access with Connection Filters
- Configuring Portlet Producers and Additional Components

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_Portal` server.

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 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 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 web tier. 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.

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 web tier'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_Portal` managed server's settings, to the web tier's host and port settings. For information about how to change these settings, see Modifying Discussions Server Connection Details.

2. Restart the `WC_Portal` managed server.

   When you log in to WebCenter Portal, you automatically sign on to the Discussion server as well.

Configuring SOA Server Connections for SSO

Assuming that you've already set up a SOA server connection, modify the URL to use the web tier host and port instead of the SOA server host and port. You can do this using Fusion

Middleware Control as described in Specifying the BPEL Server Hosting WebCenter Portal Workflows.

After modifying the URL and completing the setup required for OAM SSO, run the following command on the WebCenter Portal Administration server so that the changes take effect:

```
setBPELConnection('webcenter','WebCenter-Worklist', 'http://
webtier.example.com:7777')
```

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.

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.

Configuring the WebLogic Server Administration Console and Enterprise Manager for OAM 11g

This section describes how to optionally set up OAM 11*g* 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 web tier 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 11*g* 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`) or Enterprise Manager (`/em`).

   d. Enter a **Description** for the resource and click **Apply**.

6. Go to **Authentication Policies > Protected Resource Policy** and add the newly created resources.

7. Go to **Authorization Policies > Protected Resource Policy** and add the newly created resources.

8. In your web tier, 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.

Configuring Elasticsearch 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 Elasticsearch must be routed through the web tier OHS ports so that they can be properly authenticated (the authentication method continues to be BASIC).

For Elasticsearch to work properly, ensure that the WebCenter Content crawl URLs, which are routed through the web tier OHS ports, are excluded in the OAM policy.

To exclude the WebCenter Content Crawl URLs:

1. Log in to the Oracle Access Management Console and click **Application Security**.
2. On the Application Security page, click the **Application Domains** link in the Access Manager section.
3. On the Application Domains page, enter the name of your Application Domain and click **Search**.
4. From the Search Results, click to open your Application Domain.
5. Open the **Resources** tab and click **Create**.
6. On the Create Resources page, enter the required details:
   - **Type**: Select HTTP as the resource type.
   - **Description**: Enter a description for the resource.
   - **Host Identifier**: Select the host Identifier created while registering the WebGate agent.
   - **Resource URL**: Enter the resource URl as `/cs/idcplg`
   - **Query**: Select the **Name Value List** and click Add icon to add the values. Enter name as `IdcService` and value as `SES_CRAWLER_*`.
   - **Operations**: Select all the operations
   - **Protection Level:** Select Excluded
7. Click **Apply**.
8. To verify if the Oracle Access Management exclusion is configured properly:
   a. Log in to Oracle WebCenter Portal as administrator.
   b. Configure the Oracle HTTP Server URL for documents crawl . See Creating a Documents Crawl Source.
   c. Click **Test** to test the document crawl endpoint.

      If the OAM exclusion is configured properly, the test is successful.

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 by using Fusion Middleware Control, or the `setContentServerConnection` WLST command. For example:

```
setContentServerConnection(appName, name, webContextRoot='/cs')
```

For command syntax and examples, see setContentServerConnection in *WebCenter WLST Command Reference Reference*.

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.

Restricting Access with Connection Filters

Follow the steps below to only allow users to access WebCenter Portal and associated components through the web tier 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 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 Developing Custom Connection Filters in *Developing Applications with the WebLogic Security Service*.

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`

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, and any other producer you have explicitly configured.

Testing Your OAM Installation

After installing and configuring OAM 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.
- **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.

Configuring SAML-based Single Sign-On

Security Assertion Markup Language (SAML) enables cross-platform authentication between web-based 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.

When users are authenticated at one site that participates in a single sign-on configuration, they are automatically authenticated at other sites in the SSO configuration and do not need to log in separately.

> **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 and SAML 2.0–based single sign-on for WebCenter Portal and SOA running on different managed servers within the same domain.

Oracle WebCenter Portal supports the following SAML based Single Sign-On configurations:

- Configuring SAML1.1-based Single Sign-On
- **Configuring SAML 2.0-based Single Sign-On**

SAML Components and Topology

Figure 26-4 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 26-3 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 and discussions.

Figure 26-3 Detailed SAML Single Sign-on Components and Topology (POST Profile Configured)

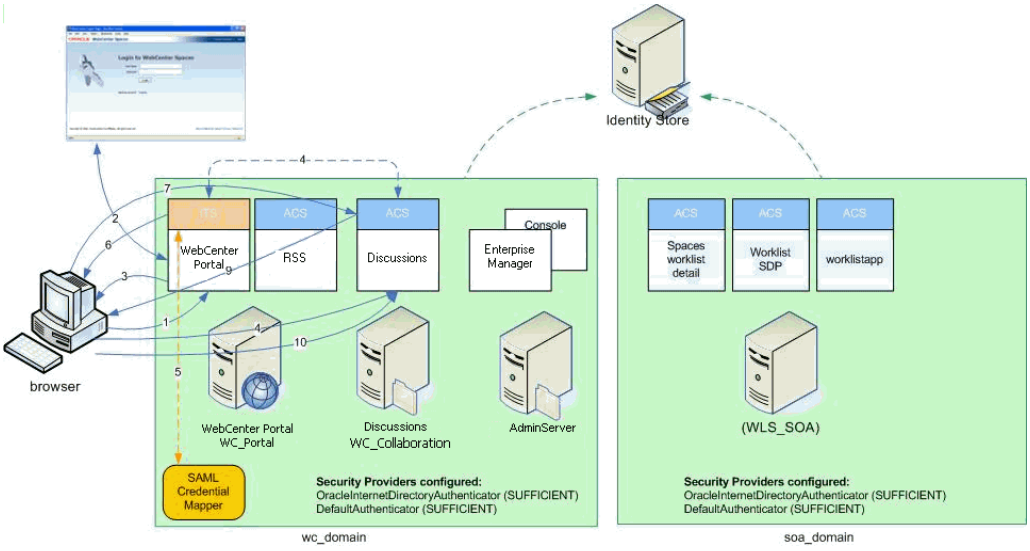
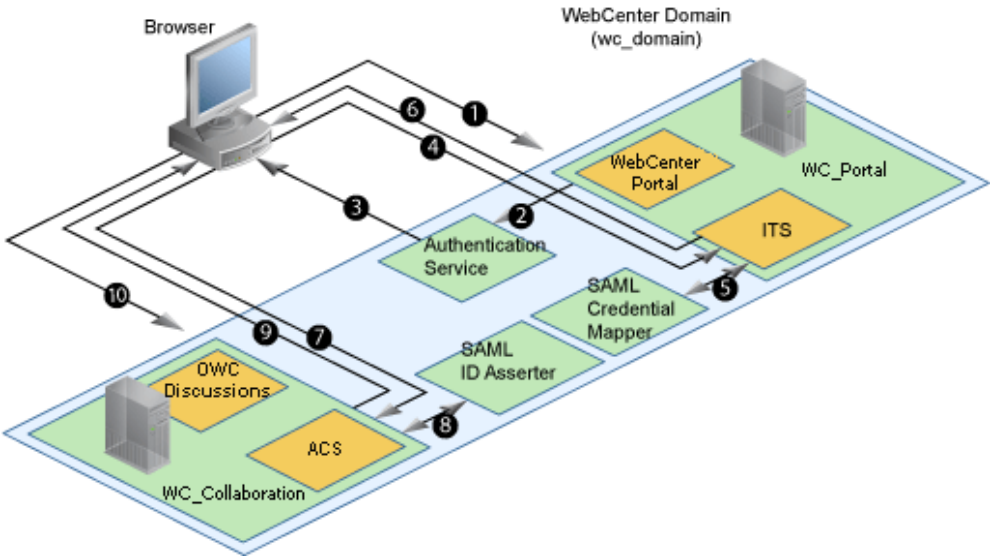


Figure 26-4 shows a simplified version of the components and flow for a POST-configured SAML SSO configuration, including the SAML SSO flow between WebCenter Portal and the discussions application.

Figure 26-4 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_Portal`) 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_Portal` 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.

Configuring SAML1.1-based Single Sign-On

This section describes how to configure WebCenter Portal and associated services and components for SAML1.1-based single sign-on using a set of automated scripts.

This section includes the following topics:

- SAML Single Sign-on Prerequisites
- Configuring SAML-based SSO
- Configuring SAML SSO for RSS Using External Readers
- Checking Your Configuration
- Disabling Your SAML SSO Configuration
- Removing Your SAML SSO Configuration

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 WebCenter Portal 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 topics:

- Configuring WebCenter Content Server for SAML SSO
- Configuring the Discussions Server for SAML SSO
- Configuring and Exporting the Certificates
- Setting Up SSL

Configuring WebCenter 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 web tier.

When configuring SAML SSO for a configuration that includes Content Server, all HTTP URLs should point to the web tier 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 Installing and Configuring 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.

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 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 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 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 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 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).

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.

Using the `getOpssService` , `listKeyStoreAliases`, and `exportKeyStoreCertificate` WLST commands, get and export the certificate you have chosen to use to encrypt SAML assertions as shown in the following example. Be sure to run the `exportKeyStoreCertificate` command on the keystore that is configured for `WC_Portal` and the Administration server for the WebCenter Portal domain. For more information, see Managing Keys and Certificates with the Keystore Service in *Securing*

Applications with Oracle Platform Security Services . For syntax for these commands, see Keystore Service Command Reference in *Securing Applications with Oracle Platform Security Services*.

The following example demonstrates how to export DemoIdentity certificate, which is available in the demoidentity keystore configured for a weblogic server by default. Use this as a guideline to list and export the certificate from the keystore configured in your environment that you wish to use for SAML configuration.

```
connect()
svc = getOpssService(name='KeyStoreService')
svc.listKeyStoreAliases(appStripe="system", name="demoidentity",
password='DemoIdentityKeyStorePassPhrase', type="*")
svc.exportKeyStoreCertificate(appStripe='system', name='demoidentity',
password='DemoIdentityKeyStorePassPhrase', alias='DemoIdentity',
type='Certificate', filepath='/tmp/demoidentity.der'
```

> **Note:**
>
> The path used in `filepath` above should match the `certPath` value in `wcsamlsso.properties`. Note also that the certificate must be exported only in PEM/DER format.

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 Configuring SSL . Note that setting up SSL is not related to enabling SSO.

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 topics:

- The Single Sign-on Script
- Using the Scripts

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
- configureSpaces.py
- configureCollab.py
- configureUtilities.py
- configureSOA.py
- configureUCM.py
- configureREST.py
- configureForum.py
- configureWorklistIntegration.py
- configureCS.py
- configureBPM.py

wcsamlsso.properties

This properties file (`WCP_ORACLE_HOME/webcenter/scripts/samlsso/wcsamlsso.properties`) encapsulates the necessary configuration information for the SAML SSO setup. Copy the properties file to the `WCP_ORACLE_HOME/common/bin` folder, change directories to that folder and edit `wcsamlsso.properties` as described below before running the configuration scripts.

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 a web tier, then specify the web tier 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` - Configuration 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` - Configuration 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` - Configuration 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 web tier, then specify the web tier 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 a web tier, then specify the web tier 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 a web tier, then specify the web tier host and port here.

`worklist_config`

Complete this section if SOA is installed and portal workflows is enabled for WebCenter Portal. For more information, see Specifying the BPEL Server Hosting WebCenter Portal Workflows.

- `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 a web tier, then specify the web tier 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 a web tier, then specify the web tier 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 web tier.

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 RSSapplication

configureForum.py

Executable script (*`WCP_ORACLE_HOME`*`/webcenter/scripts/samlsso/configureForum.py`) to configure asserting and relying parties for discussions

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.

configureBPM.py

Executable script (*WCP_ORACLE_HOME*/webcenter/scripts/samlsso/configureBPM.py) to configure asserting and relying parties for Oracle BPM Worklist.

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 configuration scripts provided are not re-runnable; you must clean up the SAML SSO artifacts before you retry the configuration as described in 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 configuration 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 configuration 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 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 SOA server connections 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/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.

`execfile('<WCP_ORACLE_HOME>/webcenter/scripts/samlsso/configureBPM.py')` - Do this if you have Oracle BPM Worklist installed in your setup. No restart is required.

7. Check your installation using the steps provided in Checking Your Configuration.

> **Note:**
>
> 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 Removing Your SAML SSO Configuration.
>
> After removing your old SAML SSO configuration, continue by re-running the scripts.

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.

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 **Portal Settings > Services > Discussions** (assuming this service is provisioned for the portal).

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_Portal` 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_Portal` and other application servers, see Viewing and Configuring WebCenter Portal Logs. Before re-running the scripts, remove your SAML SSO configuration as described in Removing Your SAML SSO Configuration.

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.

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.

Configuring SAML 2.0-based Single Sign-On

You can configure single sign-on using SAML-2.0 to enable user to sign on to an application only once and gain access to multiple applications. SAML-2.0 enables exchange of authentication information between Identity Provider and Service Provider running on the WebLogic server domain. Identity Provider acts as a source site and provides credentials for authentication. Service Provider consumes the authentication information passed by the Identity Provider.

WebLogic Server can be configured to act as a SAML Identity Provider and Service Provider. For Identity Provider, SAML credential mapping provider must be configured so that the

Identity Provider can produce assertions. For Service Provider, the SAML identity assertion provider must be configured so that the Service Provider can consume assertions.

In the configuration described in this topic, we have configured WebCenter Portal as Identity Provider and WebCenter Content as Service Provider. The Single Sign-on is being established between WebCenter Portal running on one WebLogic Server and WebCenter Content running on another WebLogic Server.

SAML 2.0 Components

- Identity Provider (IdP)—Identity Provider is a system, or administrative domain, which provides identifiers for users interacting with a system and asserts that a user has been authenticated and is given associated attributes. An Identity Provider is also known as a SAML authority, asserting party, or source site, and is often abbreviated as IdP.
- Service Provider (SP)—A system, or administrative domain, that determines whether it trusts the assertions provided to it by the Identity Provider. SAML defines a number of mechanisms that enable the Service Provider to trust the assertions provided to it. A Service Provider is also known as a relying party, or destination site, and is often abbreviated as SP.

  For example: If you want to log in to the WebCenter Content using WebCenter Portal credentials, then WebCenter Content acts as an service provider.
- Credential Mapping provider—Generates SAML 2.0 assertions. This provider must be configured for a WebLogic Server instance that serves as an Identity Provider.
- Identity Assertion provider—Consumes SAML 2.0 assertions. This provider must be configured for a WebLogic Server instance that serves as an Service Provider.
- SAML Authentication provider—Enables "virtual user" functionality SAML 2.0 Identity Assertion providers.

For more information, see Security Assertion Markup Language (SAML) in *Understanding Security for Oracle WebLogic Server*.

Prerequisites

- Installed `webcenter.ear` comes with cookie-path set with */webcenter*. Due to the imitation of WebLogic Server SAML 2.0, cookie-path must be set to /. This is required because WebLogic Service Provider supports only / as cookie-path for SAML 2.0. For more information, see Configuring a Service Provider Site for SAML 2.0 Single Sign-On.
- In case your IdP and SP are installed on the same machine or running on the same domain, and you try to log in to the IdP first and then log into the SP, the cookie-path / established during IdP login is overridden by SAML 2.0 when you try to log in to SP. Hence, the IDP session times out and you must log in again to the IdP. As a workaround for this issue, create virtual hosts for both SP and IdP and register these virtual hosts in the IdP and SP WebLogic Server configuration. In this document, virtual hosts are created using OHS. For more information, see https://httpd.apache.org/docs/2.2/vhosts/examples.html.

Main steps

A summary of the main steps you take to configure SAML 2.0 services is as follows:

1. Configuring a SAML 2.0 Identity Provider site. In this configuration, WebCenter Portal is configured as Identity Provider site.
    a. Create and configure an instance of the SAML 2.0 Credential Mapping provider. For more information, see Creating SAML 2.0 Credential Mapping Provider .
    b. Configure the SAML 2.0 Identity Provider services. See Configuring SAML 2.0 Identity Provider Services.
    c. Configure the SAML 2.0 general services and publish the metadata file. For more information, see Configure SAML 2.0 General Services for Identity Provider.
    d. Create and configure your Service Provider partners. For more information, see Configuring Service Provider Partner Metadata on SAML Identity Provider Source Site.
2. Configuring a SAML 2.0 Service Provider site. In this configuration, WebCenter Content is configured as Service Provider site.
    a. Create and configure an instance of the SAML 2.0 Identity Assertion provider. For more information, see Creating SAML 2.0 Identity Assertion Provider.
    b. Configure the SAML 2.0 Service Provider services. For more information, see Configuring SAML 2.0 Service Provider Services.
    c. Configure the SAML 2.0 general services and publish the metadata file. For more information, see Configuring SAML 2.0 General Services for Service Provider.
    d. Create and configure your Identity Provider partners. For more information, see .Configuring Identity Provider Metadata on SAML Service Provider

For more information, see Configuring SAML 2.0 Services in *Administering Security for Oracle WebLogic Server*.

This section includes the following topics:

- Creating SAML 2.0 Credential Mapping Provider
- Configuring SAML 2.0 Identity Provider Services
- Configure SAML 2.0 General Services for Identity Provider
- Configuring Service Provider Partner Metadata on SAML Identity Provider Source Site
- Creating SAML 2.0 Identity Assertion Provider
- Configuring SAML 2.0 Service Provider Services
- Configuring SAML 2.0 General Services for Service Provider
- Configuring Identity Provider Metadata on SAML Service Provider

Creating SAML 2.0 Credential Mapping Provider

You have to configure Credential Mapping Provider for WebLogic Server instance that serves as an Identity Provider. Credential Mapping Provider allows the WebLogic Server to log into a remote system that has been authenticated on your behalf. You need to configure the Credential Mapping Provider on the source site, for this example it is configured on the WebCenter Portal.

To create a SAML 2.0 Credential Mapping Provider:

1. Log in to the source domain WebLogic Server Administration Console as an administrator.

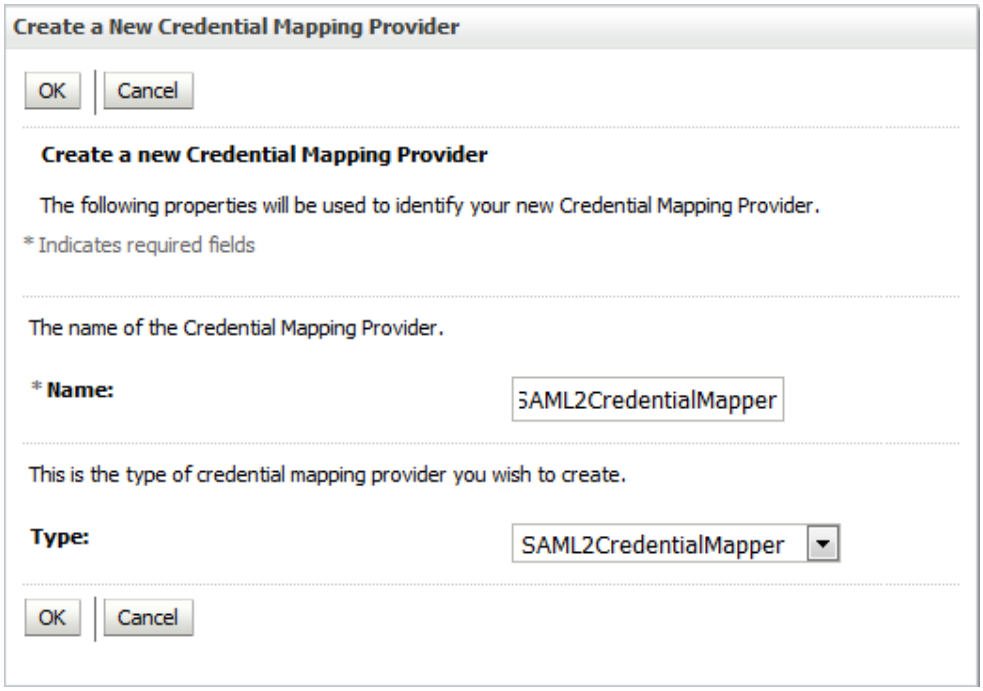
For information on logging in to the WebLogic Server Administration Console, see Oracle WebLogic Server Administration Console.

2. On the Domain Structure pane, click **Security Realms** and select `myrealm`.
3. On the Settings for *myrealm* page, click the **Providers** tab, then the **Credential Mapping** tab.

   The Credential Mapping Providers table lists the Credential Mapping providers configured in this security realm.
4. Click **New**.

   The **Create a New Credential Mapping Provider** page appears.

Figure 26-5 Creating Credential Mapping Provider

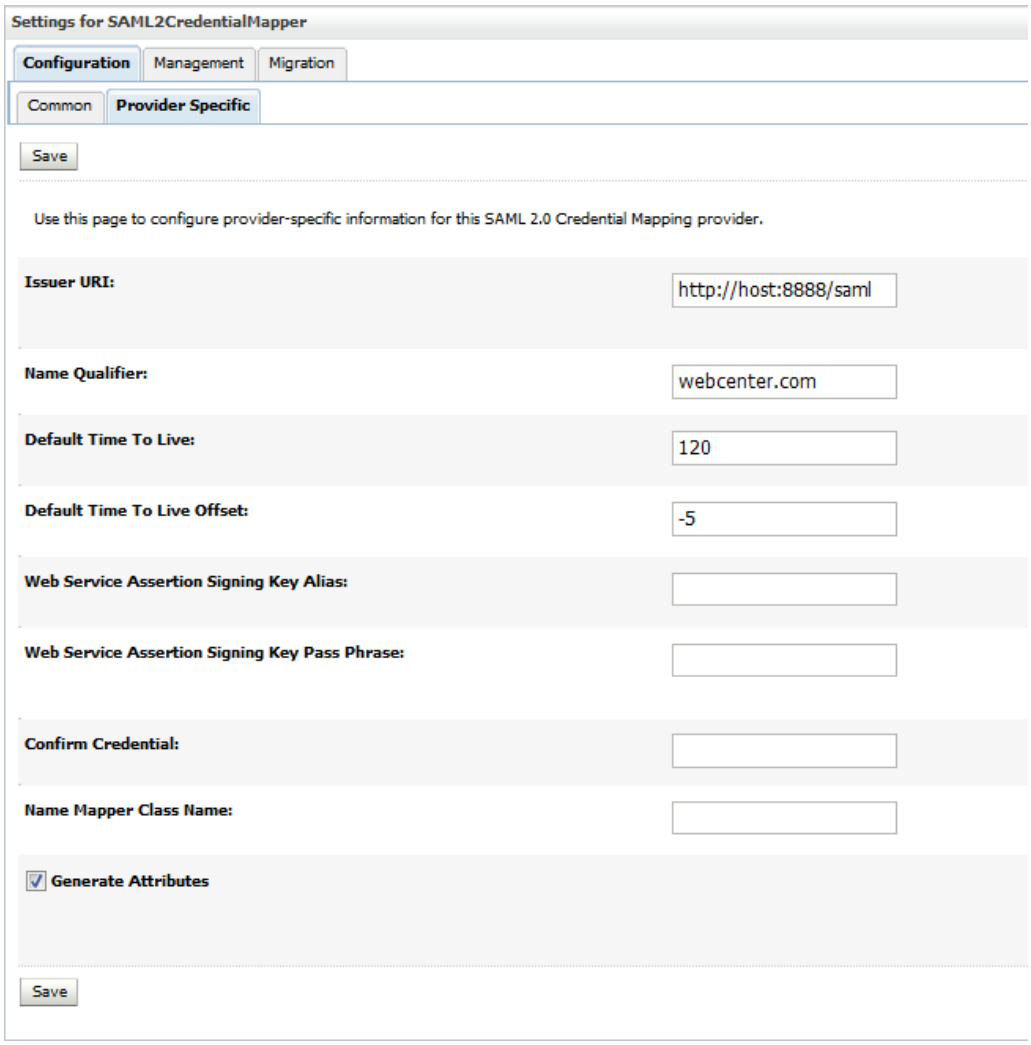


5. In the **Name** field, enter a name for the Credential Mapping Provider.

   For example, `SAML2CredentialMapper`.
6. From the **Type** drop-down list, select `SAML2CredentialMapper` and click **OK**.
7. On the Settings for *myrealm* page, select the **Providers** tab, then the **Credential Mapping** tab.
8. Click the name of the new Credential Mapping Provider to complete the configuration. For example, `SAML2CredentialMapper`.
9. Click the **Provider Specific** tab.

   The Provider Specific Settings pane for the newly added Credential Mapping Provider appears.

Figure 26-6 Configuration Settings for SAML 2.0 Credential Mapping Provider



10. Configure the provider-specific information for the newly added SAML 2.0 Credential Mapping Provider . Leave the rest of the fields set to their default values.
    - **Issuer URI** : Enter the IDP URL (http://*host:port*/saml) .
    - **Name Qualifier**: Enter *webcenter.com*
11. Click **Save** to save your changes.
12. Stop and restart all the servers..

Next **Configure Identity Providers** as described in Configuring SAML 2.0 Identity Provider Services.

Configuring SAML 2.0 Identity Provider Services

You can configure WebCenter Portal running on a Weblogic server to act as a Identity Provider Service to enable single sign-on using SAML 2.0.

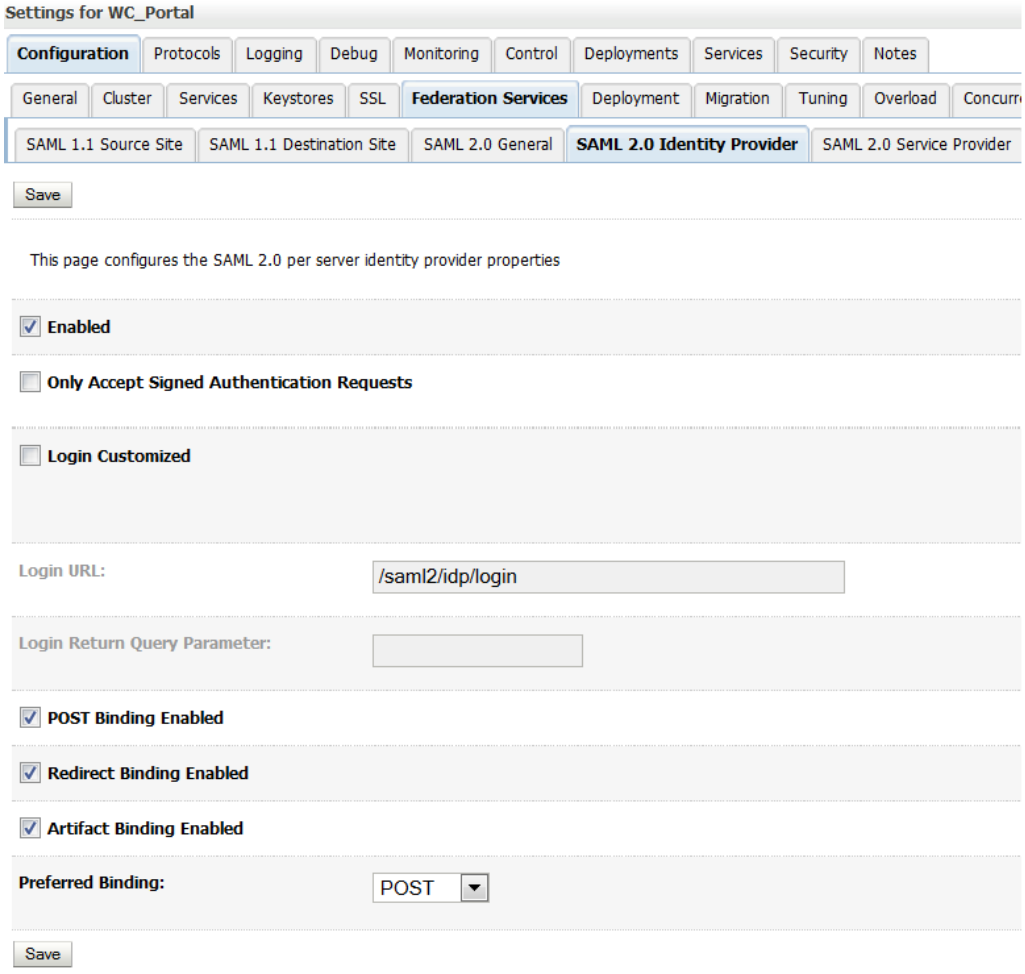
To Configure the SAML 2.0 Identity Provider services:

1. Log in to the source site WebLogic Server Administration Console as an administrator.

2. On the Home page, select **Servers** under **Environment**.
3. From the **Servers** table, select WebCenter Portal server (`WC_Portal`).
4. Click the **Federation Services** tab, then the **SAML 2.0 Identity Provider** tab.

   The **SAML 2.0 Identity Provider** page appears.
5. On the SAML 2.0 Identity Provider page, set the configuration options for the SAML 2.0 Service Provider services as appropriate.
   a. Select **Enabled** to activate SAML 2.0 services in WebCenter Portal server.
   b. From the **Preferred Binding** list, select `POST`.

Figure 26-7 Configuration Settings for SAML 2.0 Identity Provider



6. Click **Save**.

Next **Configure SAML 2.0 general services for Identity Provider**, as described in Configure SAML 2.0 General Services for Identity Provider.

Configure SAML 2.0 General Services for Identity Provider

To configure the general services for the Identity Provider:

1. On the WebLogic Server Administration Console Home page, select **Servers** under **Environment**.
2. From the **Servers** table, select WebCenter Portal server (`WC_Portal`).
3. Click the **Federation Services** tab, then the **SAML 2.0 General** tab.
4. Configure the general setting for Identity Provider as shown in the table. Leave the rest of the fields set to their default values.

Table 26-2 General Setting Parameters

Parameter	Description
Replicated Cache Enabled	Select Replicated Cache Enabled to use the persistent cache for storing SAML 2.0 artifacts. This option is required if you are configuring SAML 2.0 services in two or more WebLogic Server instances in your domain. For example, if you are configuring SAML 2.0 services in a cluster, you must enable this option in each Managed Server instance individually. The replicated cache enables server instances to share and be synchronized with the data that is managed by the SAML 2.0 security providers; that is, either or both the SAML 2.0 Identity Assertion provider and the SAML 2.0 Credential Mapping provider.
Site Info	The site information is for the benefit of the business partners in the SAML federation with whom you share it. Site information includes details about the local contact person who is your partners' point of contact, your organization name, and your organization's URL. Enter the following site information: • Contact Person Given Name • Contact Person Surname • Contact Person Type • Contact Person Company • Contact Person Telephone Number • Contact Person Email Address • Organization Name • Organization URL

Table 26-2 (Cont.) General Setting Parameters

Parameter	Description
Published Site URL	The Published site URL specifies the base URL that is used to construct endpoint URLs for the SAML 2.0 services. The published site URL should specify the host name and port at which the server is visible externally, which might not be the same at which the server is accessed locally. For example, if SAML 2.0 services are configured in a cluster, the host name and port may correspond to the load balancer or proxy server that distributes client requests to the Managed Servers in that cluster. The published site URL should be appended with `/saml2`. For example: `host:port/saml2`
Entity ID	The entity ID is a human-readable string that uniquely distinguishes your site from the other partner sites in your federation. When your partners need to generate or consume an assertion, the SAML 2.0 services use the entity ID as part of the process of identifying the partner that corresponds with that assertion. Enter Entity ID for Identity Provider as `webcenter_IDP`.
Recipient Check Enabled	Enable the Recipient Check Enabled. The recipient of the authentication request or response must match the URL in the HTTP Request.
Single Sign-on	The keystore alias and passphrase for the key is used when signing documents sent to your federated partners, such as authentication requests or responses. Enter the following information: • Single Sign-on Signing Key Alias • Single Sign-on Signing Key Pass Phrase: **Note**: In this example, OOTB WebLogic Server shipped DemoIdentity keystore is used and the password is *DemoIdentityPassPhrase*.

5. Click **Save**.
6. Click **Publish Meta Data** to create or update the partner metadata file, which contains the information about this site SAML 2.0 services to be shared with your federated partners that is used for SAML 2.0 web single sign-on.

   The **Publish SAML 2.0 Meta Data** page opens.
7. On the Publish SAML 2.0 Metadata page, enter the full path of the XML metadata file.

   For example, */mydomain/myserver/idp_metadata.xml*

> **Note:**
>
> When you are publishing the metadata file for Identity Provider, name the file as *idp_metadata.xm1*

8. Click **OK** to publish the metadata file.

   The metadata file is published and copied to the specified path.

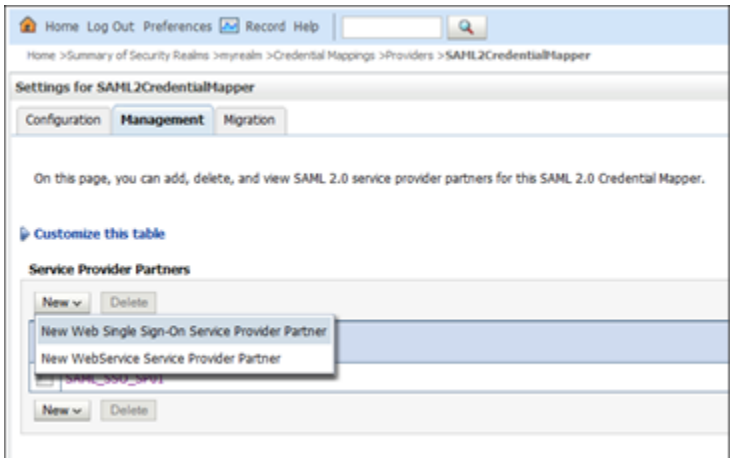
Next **Configure Service Provider Metadata on SAML Identity Provider Source Site**, as described in Configuring Service Provider Partner Metadata on SAML Identity Provider Source Site

Configuring Service Provider Partner Metadata on SAML Identity Provider Source Site

To create a SAML 2.0 Service Provider partner metadata on the source server:

1. In the WebLogic Server Administration Console, click **Security Realms** and select `myrealm`.
2. Click the **Providers** tab, then the **Credential Mapper** tab
3. Select the SAML 2.0 Credential Mapping provider (For example, *SAML2CredentialMapper*).
4. On the Settings for SAML 2.0 Credential Mapping Provider page, click the **Management** tab.
5. Under **Service Provider Partners**, click **New** and select **New Web Single Sign-On Service Provider Partner**.
6. On the Create a SAML 2.0 Web Single Sign-on Service Provider Partner page:

Figure 26-8 New Web Single Sign-On Service Provider Partner



   a. Enter the name of the Service Provider partner.

      For example *SAML_SSO_SP01*

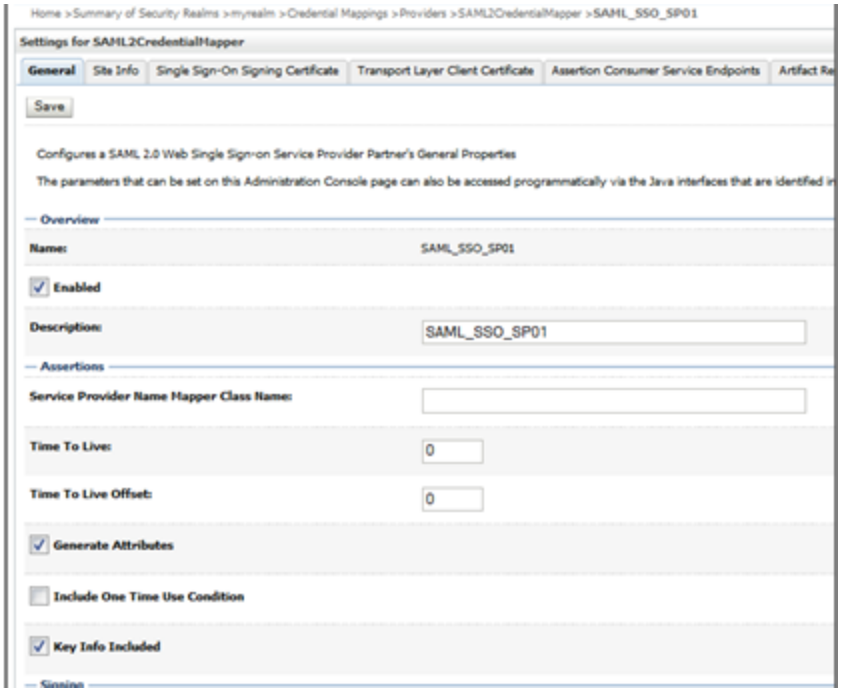
b. In the field next to **Path**, specify or browse to the full path of the metadata partner file.

   For example, *sp_metadata.xml* file.

7. Click **OK**.
8. On the Settings for SAML 2.0 Credential Mapper page, in the **Service Provider Partners** table, select the name of your newly-created Service Provider partner.

   For example, *SAML_SSO_SP01*.
9. On the General page, configure the following settings as appropriate:

Figure 26-9 SAML 2.0 Web Single Sign-on Service Provider Partner General Settings



a. Select **Enabled** to enable interactions between this server and this Service Provider partner.

b. In the **Description** field, enter the description of the Service Provider partner. For example, *SAML_SSO_SP01.*

c. Select **Key Info Included**

10. Click **Save**.

    The Service Provider partner is created in the local server instance.

Creating SAML 2.0 Identity Assertion Provider

You can configure SAML 2.0 Identity Assertion provider to act as a consumer of SAML 2.0 security assertions, allowing WebLogic Server to act as a Service Provider for web

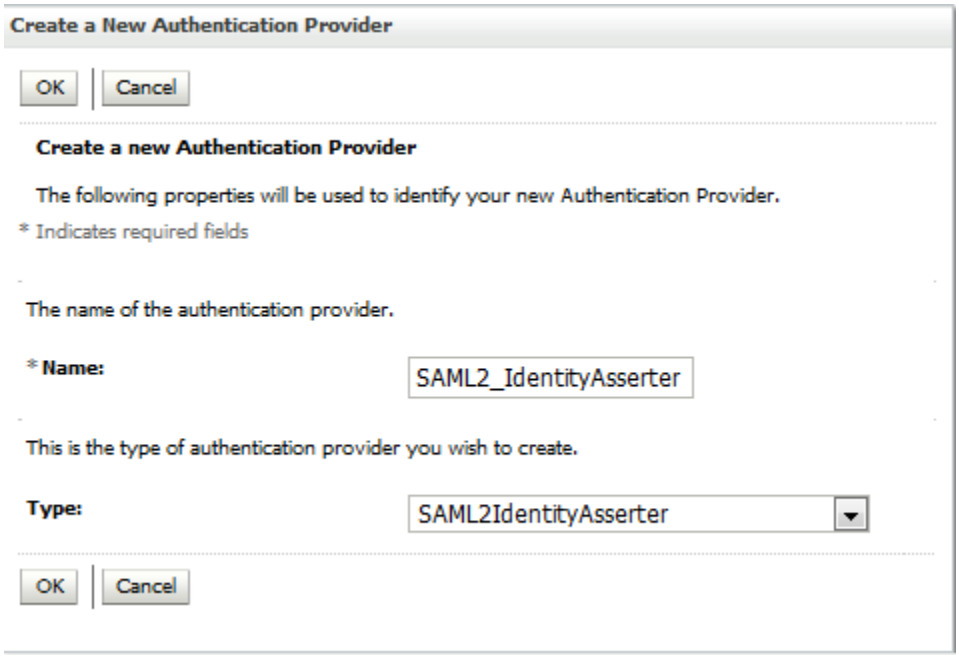
single sign-on. You need to configure the Identity Assertion provider on the destination site, for this example it is configured on the WebCenter Content.

To create SAML 2.0 Identity Assertion Provider in the destination domain

1. Log in to the destination site WebLogic Server Administration Console as an administrator.
2. On the Domain Structure pane, click **Security Realms** and select `myrealm`.
3. On the Settings for *myrealm* page, click the **Providers** tab, then the **Authentication** tab.
4. Click **New**.

   The **Create a New Authentication Provider** page appears.

Figure 26-10 Creating Authentication Provider



5. In the **Name** field, enter a name for the Authentication provider. For example, `SAML2_IdentityAsserter`
6. From the **Type** drop-down list, select `SAML2_IdentityAsserter`
7. Click **OK**
8. Stop and restart all the servers.

Next **Configure the SAML 2.0 Service Provider services** as described in Configuring SAML 2.0 Service Provider Services.

Configuring SAML 2.0 Service Provider Services

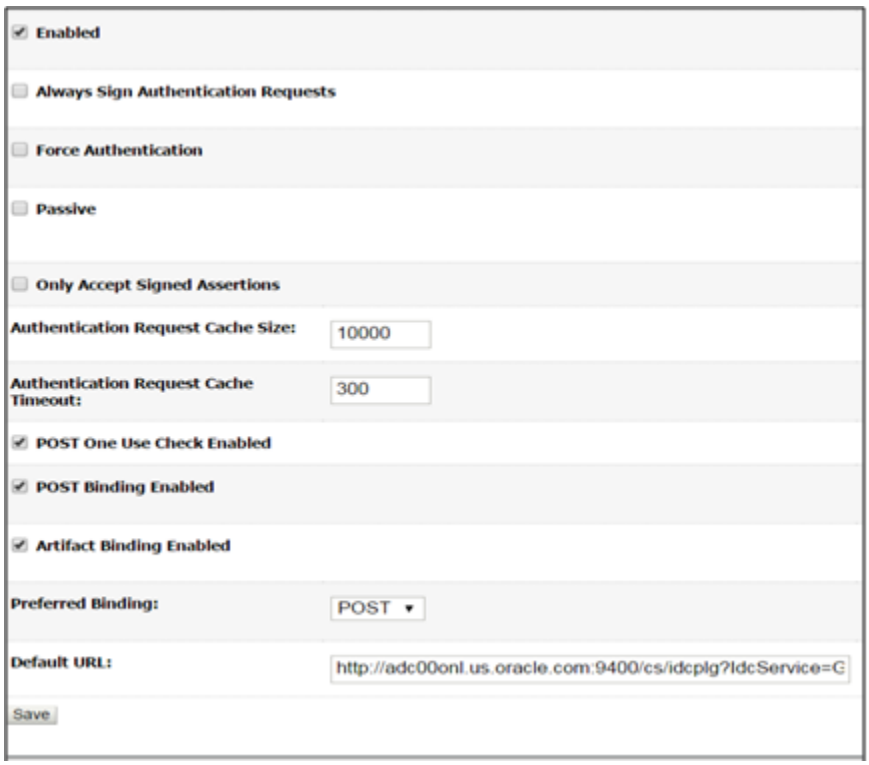
To configure a server as a SAML 2.0 Service Provider:

1. Log in to the destination site WebLogic Server Administration Console as an administrator.
2. On the Home page, select **Servers** under **Environment**.

3. From the **Servers** table, select WebCenter Content server (`UCM_server1`).
4. Click the **Federation Services** tab, then **SAML 2.0 Service Provider** tab.

   The **SAML 2.0 Service Provider** page appears.
5. On the SAML 2.0 Identity Service page, set the configuration options for the SAML 2.0 Service Provider services as appropriate.
   a. Select **Enabled** to activate SAML 2.0 services in WebLogic server in the role of Service Provider.
   b. From the **Preferred Binding** list, select `POST`.
   c. In the Default URL field, enter the destination URL.`http://host:port/cs/idcplg?IdcService=GET_DOC_PAGE&Action=GetTemplatePage&Page=HOME_PAGE&Auth=Internet`

Figure 26-11 Configuration Settings for SAML 2.0 Service Provider



6. Click **Save**.

Next **Configure SAML 2.0 general services for service provider**, as described in Configuring SAML 2.0 General Services for Service Provider.

Configuring SAML 2.0 General Services for Service Provider

To configure the general services for SAML 2.0:

1. On the WebLogic Server Administration Console Home page, select **Servers** under **Environment**.
2. From the **Servers** table, select WebCenter Content server (`UCM_server1`).

3. Click the **Federation Services** tab, then the **SAML 2.0 General** tab.
4. Configure the general settings for service provider site as shown in the table. Leave the rest of the fields set to their default values.

Table 26-3 General Setting Parameters

Parameter	Description
Replicated Cache Enabled	Select Replicated Cache Enabled to use the persistent cache for storing SAML 2.0 artifacts. This option is required if you are configuring SAML 2.0 services in two or more WebLogic Server instances in your domain. For example, if you are configuring SAML 2.0 services in a cluster, you must enable this option in each Managed Server instance individually. The replicated cache enables server instances to share and be synchronized with the data that is managed by the SAML 2.0 security providers; that is, either or both the SAML 2.0 Identity Assertion provider and the SAML 2.0 Credential Mapping provider.
Site Info	The site information is for the benefit of the business partners in the SAML federation with whom you share it. Site information includes details about the local contact person who is your partners' point of contact, your organization name, and your organization's URL. Enter the following site information: • Contact Person Given Name • Contact Person Surname • Contact Person Type • Contact Person Company • Contact Person Telephone Number • Contact Person Email Address • Organization Name • Organization URL
Published Site URL	The Published site URL specifies the base URL that is used to construct endpoint URLs for the SAML 2.0 services. The published site URL should specify the host name and port at which the server is visible externally, which might not be the same at which the server is accessed locally. For example, if SAML 2.0 services are configured in a cluster, the host name and port may correspond to the load balancer or proxy server that distributes client requests to the Managed Servers in that cluster. The published site URL should be appended with `/saml2`. For example: `host:port/saml2`

Table 26-3 (Cont.) General Setting Parameters

Parameter	Description
Entity ID	The entity ID is a human-readable string that uniquely distinguishes your site from the other partner sites in your federation. When your partners need to generate or consume an assertion, the SAML 2.0 services use the entity ID as part of the process of identifying the partner that corresponds with that assertion. Enter Entity ID for Service Provider as `webcenter_SP`
Recipient Check Enabled	Enable the Recipient Check Enabled. The recipient of the authentication request or response must match the URL in the HTTP Request.
Single Sign-on	The keystore alias and passphrase for the key is used when signing documents sent to your federated partners, such as authentication requests or responses. Enter the following information: • Single Sign-on Signing Key Alias • Single Sign-on Signing Key Pass Phrase: **Note**: In this example, OOTB WebLogic Server shipped DemoIdentity keystore is used and the password is *DemoIdentityPassPhrase*.

5. Click **Save**.
6. Click **Publish Meta Data** to create or update the partner metadata file, which contains the information about this site's SAML 2.0 services to be shared with your federated partners that is used for SAML 2.0 web single sign-on.

   The **Publish SAML 2.0 Meta Data** page opens.
7. On the Publish SAML 2.0 Metadata page, enter the full path of the XML metadata file.

   For example, */mydomain/myserver/sp_metadata.xml*

   > **Note:**
   >
   > When you are publishing the metadata file for Service Provider, name the file as *sp_metadata.xm1*
8. Click **OK** to publish the metadata file.

   The metadata file is published and copied to the specified path.

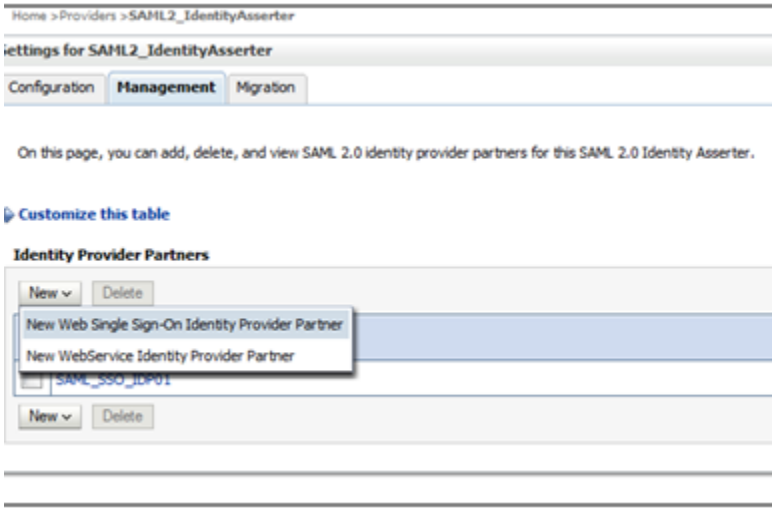
Next **Create and configure your Identity Provider partners** on the destination server, as described in Configuring Identity Provider Metadata on SAML Service Provider

Configuring Identity Provider Metadata on SAML Service Provider

To create a SAML 2.0 Identity Provider partner on the destination server:

1. In the destination site WebLogic Server Administration Console, click **Security Realms** and click`myrealm`.
2. On the Settings for *myrealm* page, click the **Providers** tab, then the **Authentication** tab.
3. In the Authentication Providers table, select the SAML 2.0 Identity Assertion provider (for example, `SAML2_IdentityAsserter`).
4. On the Settings for SAML 2.0 Identity Asserter page, click the **Management** tab.
5. Under **Identity Provider Partners**, click **New** and select **New Web Single Sign-On Identity Provider Partner**.

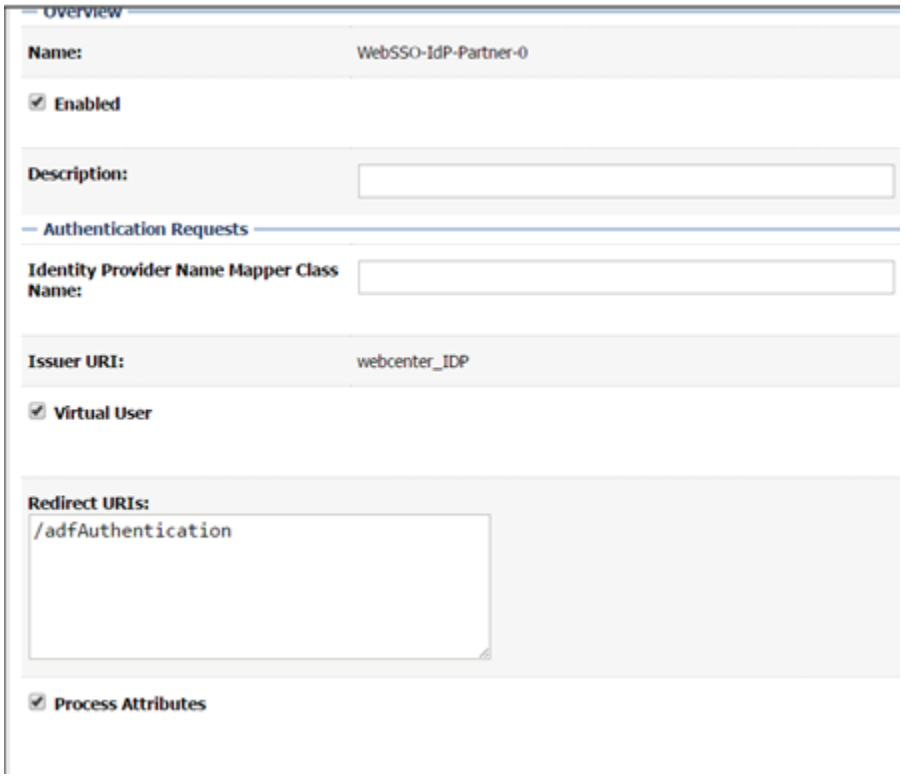
Figure 26-12 New Web Single Sign-On Identity Provider Partner



6. On the Create a SAML 2.0 Web Single Sign-on Identity Provider Partner page:
   a. Specify the name of the name of the New Web Single Sign-on Identity Provider partner. For example, `WebSSO-IdP-Partner-0`
   b. In the field next to **Path**, specify or browse the name and location of the SAML 2.0 metadata file received from the Identity Provider partner. For example, *idp_metadata.xml* file.
7. Click **OK**.
8. On the Settings for SAML 2.0 Identity Asserter page, in the **Identity Provider Partners** table select the name of your newly-created web single sign-on Identity Provider partner.

   For example: `WebSSO-IdP-Partner-0`
9. On the General page, configure the following configure the following settings as appropriate:

Figure 26-13 SAML 2.0 Web Single Sign-on Identity Provider Partner General Settings



a. Select **Enabled** to enable interactions between this server and this Identity Provider partner.

b. Enter a short description of this Identity Provider partner.

c. Select **Virtual User** to specify user information contained in assertions received from this Identity Provider partner are mapped to virtual users in this security realm.

d. In the **Redirect URIs** field, specify the URIs for resources hosted at the local site that, if invoked by an unauthorized user, cause an authentication request to be generated and sent to the Identity Provider partner. For example, */adfAuthentication* for content server.

10. Click **Save.**

Troubleshooting Common Issues with SAML 2.0

This section provides information to assist you in troubleshooting the problems you may encounter while configuring SAML 2.0 based Single Sign-On.

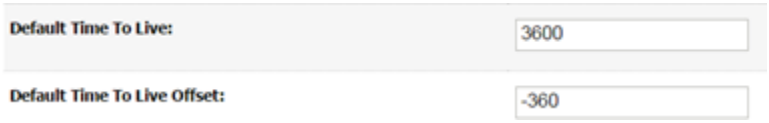
If there is difference in the time between the Identity Provider and Service Provider, the SSO will not be established.

For example, if the Service Provider time was set one minute behind the Identity Provider, the following error appears, when you access the Service Provider instance:

```
<Sep 2, 2015 1:08:28 AM EDT> <Debug> <SecuritySAML2Service> <BEA-000000>
<[Security:090377]Identity Assertion Failed,
weblogic.security.spi.IdentityAssertionException: [Security:090377]Identity
Assertion Failed, weblogic.security.spi.IdentityAssertionException:
[Security:096537]Assertion is not yet valid (NotBefore condition).>
```

Ensure the Identity Provider and Service Provider is synchronized. We recommend you to adjust the default values of **Default Time to Live** and **Default Time to Live Offset** to fix the offset in the timings between the Identity Provider and Service Provider.

Figure 26-14 Setting the Default Time



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 25, "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:

- Microsoft Client SSO Concepts
- System Requirements
- Configuring Microsoft Clients

Microsoft Client SSO Concepts

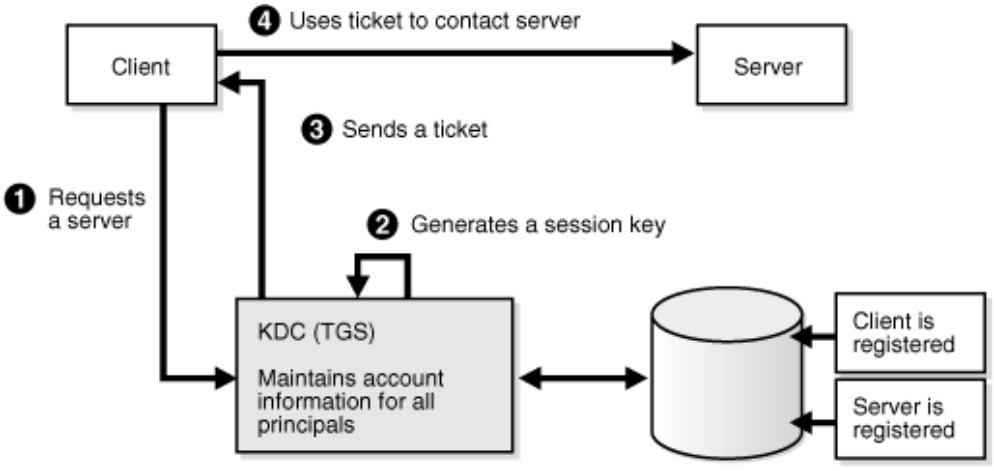
Understanding Kerberos

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 26-15 Connecting to a Server Through a Key Distribution Center



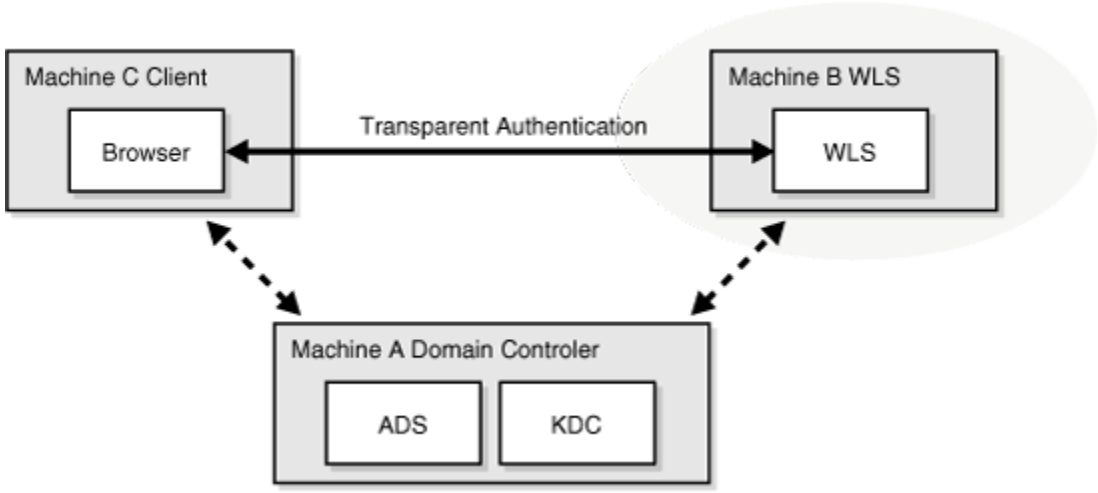
Understanding SPNEGO

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 submechanisms 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 26-16 SPNEGO-based Authentication



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:

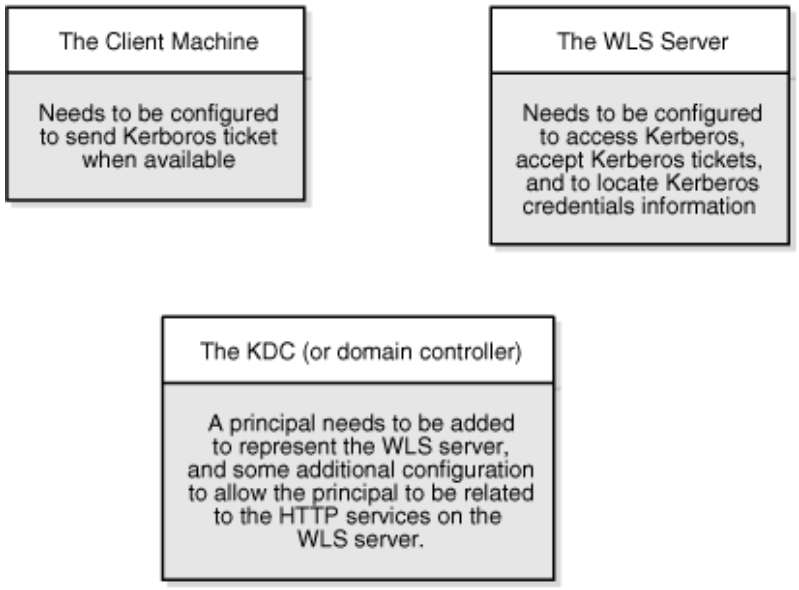
- Windows 2000 Professional SP2 or later installed
- One of the following types of clients:
  - A properly configured Internet Explorer browser. Internet Explorer 6.01 or later is supported.
  - .NET Framework 1.1 and a properly configured Web service client.

> **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 logins will not work.

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 26-17. For detailed configuration steps and troubleshooting, see Configuring Single Sign-On with Microsoft Clients in *Administering Security for Oracle WebLogic Server*.

Figure 26-17 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 Configuring Single Sign-On with Microsoft Clients in *Administering Security for Oracle WebLogic Server*).
3. Choose a browser client (Internet Explorer or Mozilla Firefox) and configure it to use Kerberos tokens (see “Enabling the Browser to Return Kerberos Tokens” in *Oracle Argus Insight Installation Guide*).
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 in *Administering Security for Oracle WebLogic Server*).

b. Configure a Negotiate Identity Assertion provider in the WebLogic Server security realm (see 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 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_Portal server without going through OHS)
-Dnon_sso_host=example.com (the host for the WLS WC_Portal server)
-Dnon_sso_port=8888 (the port for the WLS WC_Portal server)
-Dsso_base_url=http://example.com:7777 (the URL for accessing the WC_Portal 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 web tier OHS so that it forwards requests to the Oracle WebLogic Server for WebCenter Portal, as described in 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 Configuring the Discussions Server for SSO).

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 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**.

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 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 26-4 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:

    a. Negotiate Identity Asserter

    b. ActiveDirectoryAuthenticator (`SUFFICIENT`)

    c. DefaultAuthenticator (`SUFFICIENT`)

    d. Other authenticators...

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_Portal` 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.

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 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`.

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:

- Understanding the Need for a Virtual Host
- Configuring Virtual Hosts for OAM 11g

Understanding 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.

Configuring Virtual Hosts for OAM 11g

To configure OAM 11g for virtual hosts requires bypassing 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 Configuring Oracle Access Manager.

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.

27
Configuring SSL

Secure WebCenter Portal and components with SSL.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

> **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 Understanding Administrative Operations, Roles, and Tools.

> **Note:**
>
> The following can use WS-Security with message protection, and consequently have no hard requirement for SSL:
>
> - BPEL servers - Oracle BPM Worklist
> - WSRP Producers
> - Discussions and announcements

Topics:

- Securing the Browser Connection to WebCenter Portal using SSL
- Securing the Connection from Oracle HTTP Server to WebCenter Portal with SSL
- Securing the Browser Connection to Discussions with SSL
- Securing the WebCenter Portal Connection to Portlet Producers with SSL
- Securing the WebCenter Portal Connection to the LDAP Identity Store
- Securing the WebCenter Portal Connection to IMAP and SMTP with SSL
- Securing the WebCenter Portal Connection to an External BPEL Server with SSL

- Securing the WebCenter Portal Connection to Content Server with SSL

Securing the Browser Connection to WebCenter Portal using SSL

This section presents an overview of how to configure the Oracle Platform Security Services (OPSS) Keystore Service for use with WebCenter Portal. It is possible to use Fusion Middleware Control as well for this, but the scope of this document is restricted to usage of WLST.

> **Note:**
>
> The default Java Keystore Service (JKS) has been replaced with the Oracle Platform Security Services (OPSS) Keystore Service. Use `WC_Portal` as the server and OPSS as the keystore service.
>
> For detailed information and step-by-step instructions to configure SSL in the WebLogic Server environment, see Managing Keys and Certificates with the Keystore Service in *Securing Applications with Oracle Platform Security Services*.

The OPSS Keystore Service provides an alternate mechanism to manage keys and certificates for message security. The OPSS Keystore Service makes using certificates and keys easier by providing central management and storage of keys and certificates for all servers in a domain. You use the OPSS Keystore Service to create and maintain keystores of type `KSS`.

Securing the browser connection to WebCenter Portal with SSL consists of the following steps

- Creating the Custom Keystore
- Configuring the Custom Identity and Custom Trust Keystores
- Configuring the SSL Connection

Creating the Custom Keystore

The first step is to generate a custom keystore for WebCenter Portal.

To configure the Keystore Service:

1. Connect to WebLogic Server using the WLST console:

   ```
   connect('username','password','hostname:port')
   ```

2. Get the OPSS Service reference name.

   ```
   svc = getOpssService(name='KeyStoreService')
   ```

3. Create a new keystore:

> **Note:**
>
> Create a keystore in the system stripe and the permission must be set to false.

Enter the following command:

```
svc.createKeyStore(appStripe='system', name='webcenter_wls', password='password',
permission=false)
```

where:

- *svc* = the service command object obtained through a call to `getOpssService()`
- *appStripe* = the name of the stripe in which the keystore is created
- *name* = the name of the keystore
- *password* = password of the keystore
- *permission* = false if protected by both permission and password (true if keystore is protected by permission only)

4. Generate key pair.

   Use your actual alias, domain name, and credentials. The following example also uses a default CA signed certificate.

   ```
   svc.generateKeyPair(appStripe='system', name='webcenter_wls', password='password',
   dn='cn=webcenteridentity,dc=example,dc=com', keysize='2048',
   alias='webcenter_wls',  keypassword='password')
   ```

   where:

   - *svc* = the service command object obtained through a call to `getOpssService()`
   - *appStripe* = the name of the stripe containing the keystore
   - *name* = the name of the keystore where the key pair is generated
   - *password* = password of the keystore
   - *dn* = the distinguished name of the certificate wrapping the key pair
   - *keysize* = the key size
   - *alias* = the alias of the key pair entry
   - *keypassword* = the key password

5. (Optional) List the keystores and aliases inside the keystore, using the following command:.

   ```
   svc.listKeyStores(appStripe='*')
   ```

   This will list the `system/webcenter_wls`.

   where:

   - *svc* = the service command object obtained through a call to `getOpssService()`
   - *appStripe* = the name of the stripe whose keystores are listed

   Enter:

   ```
   svc.listKeyStoreAliases(appStripe="system",name="webcenter_wls",
   password="password", type="*")
   ```

This will list the alias `webcenter_wls`.

where:

- *svc* = the service command object obtained through a call to `getOpssService()`
- *appStripe* = the name of the stripe containing the keystore
- *name* = the name of the keystore
- *password* = password of the keystore
- *type* = the type of entry for which aliases are listed. Valid values are `'Certificate'`, `'TrustedCertificate'`, `'SecretKey'` or `'*'`

6. Run the `syncKeyStores` command:

```
syncKeyStores(appStripe='system', keystoreFormat='KSS')
```

7. Restart the `WC_Portal` managed server.

Configuring the Custom Identity and Custom Trust Keystores

For an overview of on how to configure the Identity and Trust keystores, see Configuring the OPSS Keystore Service for Custom Identity and Trust: Main Steps in *Administering Security for Oracle WebLogic Server*.

The next step is to configure the Custom Identity and Custom 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 Oracle WebLogic Server Administration Console.

2. Click the WebCenter Portal server (`WC_Portal`) to configure the identity and trust keystores.

   The **Settings** pane for the WebCenter Portal server opens.

3. Open the **Configuration** tab, and then the **Keystores** subtab.

   The **Keystores** pane opens.

4. Click **Change**.

5. For **Keystores**, select `Custom Identity and Custom Trust` and click **Save**.

6. Under **Identity**, enter the path and filename of the Custom Identity Keystore you created in Securing the Browser Connection to WebCenter Portal using SSL.

   If you use the example in Securing the Browser Connection to WebCenter Portal using SSL, enter `kss://system/webcenter_wls`
   where

   - *alias* = `system`
   - *keystore_alias_name* = `webcenter_wls`

7. Enter `KSS` as the **Custom Identity Keystore Type**.

8. Enter and confirm the Custom Identity Keystore password.

9. Under Trust, set the **Custom Trust Keystore** to `kss://system/trust`.
10. For **Custom Trust Keystore Type**, enter `KSS`, then click **Save** to save your entries.
11. Open the SSL tab.
12. Enter the **Private Key Alias** (for example, `webcenter_wls`) and the **Private Key Passphrase** (for example, `welcome1`), then click **Save** to save your entries.

Configuring the SSL Connection

For an overview to configure the SSL connection, see Specifying a Client Certificate for an Outbound Two-Way SSL Connection in *Administering Security for Oracle WebLogic Server*.

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. Restart the WebLogic Server and open the SSL WebCenter 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 the following command:
   `-Dweblogic.security.SSL.ignoreHostnameVerification=true`
9. Accept the certificate for the session and log in.

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:

- Configuring the Custom Identity and Custom Trust Keystores
- Configuring the SSL Connection
- Wiring the WebCenter Portal Ports to the HTTP Server
- Configuring the SSL Certificates

Wiring the WebCenter Portal Ports to the HTTP Server

To wire the WebCenter Portal ports to the HTTP server:

1. Install and configure OHS 12*c* (see Installing the Oracle HTTP Server Software in *Installing and Configuring Oracle HTTP Server*).

   By default, it comes configured with the SSL port.

2. Open the file *DOMAIN_HOME*/config/fmwconfig/components/OHS/instances/ohs1/mod_wl_ohs.conf

3. Add the WebCenter URL to `mod_wl_ohs.conf` to make WebCenter Portal work with OHS:

   ```
   <Location/webcenter>
   SetHandler weblogic-handler
   WebLogicHost host_id
   WebLogicPort port
   </Location>
   ```

   Replacing *host_id* and *port* with the WebLogic server ID and port number.

   > **Note:**
   >
   > When using `mod_wl_ohs`, you need to complete the prerequisites mentioned in Preparing for Configuring the Oracle WebLogic Server Proxy Plug-In in *Using Oracle WebLogic Server Proxy Plug-Ins 12.2.1*.

4. Start the node manager:

   DOMAIN_HOME/bin/startNodeManager.sh &

   See Starting the Node Manager in *Installing and Configuring Oracle HTTP Server*.

5. Restart the OHS server:

   DOMAIN_HOME/bin/stopComponent.sh ohs1 & *DOMAIN_HOME*/bin/startComponent.sh ohs1

6. Verify if the following URLs are working:

   http://*OHS_12c_installation_host*:*port*

   http://*OHS_12c_installation_host*:*OHS_12c_installation_port*/webcenter

7. Configure the WebCenter SSL port with the OHS SSL:

   a. Verify that the OHS SSL port is working by checking the following URL.

      https://*ohs_ssl_host*:*ohs_ssl_port*

   b. To configure the WebCenter SSL port, open the file OHS `ssl.conf` file (*DOMAIN_HOME*/config/fmwconfig/components/OHS/instances/ohs1/ssl.conf.

   c. Add the following entry (WebCenter SSL host and port) to `ssl.conf` to make WebCenter Portal run on the OHS SSL port:

> **Note:**
>
> This snippet needs to be inserted just before the `</VirtualHost>` tag, that is, where the virtual host tag ends.

```
  <Location /webcenter>
SetHandler weblogic-handler
   WebLogicHost host_id
WebLogicPort port
SecureProxy ON
WlSSLWallet /filepath/ohs12c/user_projects/domains/base_domain/config/
fmwconfig/components/OHS/instances/ohs1/keystores/default
  </Location>
```

8. Restart OHS.

Configuring the SSL Certificates

For OHS to trust WebCenter Portal's certificate, the `WC_Portal` certificate must be imported into the OHS trust store.

To configure the SSL certificates:

1. Export the `WC_Portal` certificate from the `WC_Portal` identity keystore, using the following WLST:

   ```
   svc = getOpssService(name='KeyStoreService')
   svc.exportKeyStoreCertificate(appStripe='system', name='webcenter_wls',
   password='password', alias='webcenter_wls', type='TrustedCertificate', filepath='/
   filepath/certificate/webcenter.cer')
   ```

   where:

   - *svc* = the service command object obtained through a call to `getOpssService()`
   - *appStripe* = the name of the stripe containing the keystore
   - *name* = the name of the keystore
   - *password* = password of the keystore
   - *type* = the type of entry for which aliases are listed. Valid values are `'Certificate'`, `'TrustedCertificate'`, or `'CertificateChain'`
   - *filepath* = absolute path of the file where certificate, trusted certificate or certificate chain is exported

2. Import this certificate into the wallet on the OHS side.

   Navigate to `/domain_home/config/fmwconfig/components/OHS/instances/ohs1/keystores/default` and run the following `orapki` command (typically located in `IDM_HOME`) :

   ```
   setenv JAVA_HOME /Java_install_location/jdk1.8.0_40/
   /OHS_install_location/oracle_common/bin/orapki wallet add -wallet . -trusted_cert -
   cert <webcenter_wls.cer location> -auto_login_only
   ```

3. 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.

```
/OHS_install_location/oracle_common/bin/orapki wallet display -wallet .
/OHS_install_location/oracle_common/bin/orapki wallet export -wallet . -cert
cert.txt -dn 'dn_value'
```

Where, *dn_value* refers to the output returned by the `wallet display -wallet` command.

4. Import the OHS certificate into the `WC_Portal` managed server trust store:

```
keytool -importcert -alias ohs_cert -file wls_java_home/jre/lib/security/
cacerts
```

Where, *wls_java_home* refers to the WebLogic Java home directory, and `keytool` is installed in *wls_java_home*`/jre/bin/keytool`. For finding out the *wls_java_home* path, you can run *domain_home*`/bin/setDomainEnv.sh` (on UNIX) or *domain_home*`\bin\setDomainEnv.cmd` (on Windows).

5. In WebCenter, log in to the WebLogic Console and check if the WebLogic Plugin checkbox is enabled:
   a. Log in to the WebLogic Console.
   b. Click the domain name on the left hand navigation.
   c. Click the **Web Applications** tab.
   d. Select the option **WebLogic Plugin Enabled**, then click **Save**.
6. Restart OHS and the `WC_Portal` server.

   You should now be able to access the SSL OHS URL (`https://<ohs ssl host>:<ohs ssl port>/webcenter`).
7. After accessing the URL, accept the certificate.

Securing the Browser Connection to Discussions with SSL

Securing the browser connection to discussions with SSL is described in the following sections:

- Creating the Custom Keystore for Discussions
- Configuring the Identity and Trust Keystore for Discussions
- Configuring and Securing the SSL Connection for Discussions

Creating the Custom Keystore for Discussions

The first step in securing the connection to Discussions is to generate a custom keystore as shown below:

1. Connect to WebLogic Server using the WLST console:

```
connect('weblogic','password','host:port')
```

2. Get OPSS service reference:

```
svc = getOpssService(name='KeyStoreService')
```

3. Create a new keystore:

> **Note:**
>
> Create the keystore in the system stripe and the permission must be set to false

```
svc.createKeyStore(appStripe='system', name='collab_wls', password='password',
permission=false)
```

where:

- *svc* = the service command object obtained through a call to `getOpssService()`
- *appStripe* = the name of the stripe in which the keystore is created
- *name* = the name of the keystore
- *password* = password of the keystore
- *permission* = true if keystore is protected by permission only; false if protected by both permission and password

4. Using keytool, generate a key pair:

```
svc.generateKeyPair(appStripe='system', name='collab_wls', password='password',
dn='cn=collabidentity,dc=example,dc=com', keysize='2048', alias='collab_wls',
keypassword='welcome1')
```

where:

- *svc* = the service command object obtained through a call to `getOpssService()`
- *appStripe* = the name of the stripe containing the keystore
- *name* = the name of the keystore where the key pair is generated
- *password* = password of the keystore
- *dn* = the distinguished name of the certificate wrapping the key pair
- *keysize* = the key size
- *alias* = the alias of the key pair entry
- *keypassword* = the key password

5. Optionally, list the keystores and aliases inside the keystore:

```
svc.listKeyStores(appStripe='*')
```

This will list the `system/collab_wls`.

where:

- *svc* = the service command object obtained through a call to `getOpssService()`
- *appStripe* = the name of the stripe whose keystores are listed

Enter:

```
svc.listKeyStoreAliases(appStripe="system",name="collab_wls", password="password",
type="*")
```

This is will list the alias `collab_wls`

where:

- *svc* = the service command object obtained through a call to `getOpssService()`

- *appStripe* = the name of the stripe containing the keystore
- *name* = the name of the keystore
- *password* = password of the keystore
- *type* = the type of entry for which aliases are listed. Valid values are `'Certificate'`, `'TrustedCertificate'`, `'SecretKey'` or `'*'`

6. Run `syncKeyStores`:

```
syncKeyStores(appStripe='system', keystoreFormat='KSS')
```

Configuring the Identity and Trust Keystore for Discussions

The next step is to configure the Custom Identity and Custom Trust keystores on the WebCenter Collaboration server.

To configure the identity and trust keystores for discussions:

1. Log in to the WebLogic Server Administration Console.

   For information on logging into the WebLogic Server Administration Console, see 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 WebCenter 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. Click **Change**.

6. For **Keystores**, select **Custom Identity and Custom Trust**, then click **Save**.

7. Under **Identity**, enter the path and filename of the Custom Identity Keystore you created in `kss://system/collab_wls` (Creating the Custom Keystore for Discussions).

8. Enter `KSS` as the **Custom Identity Keystore Type**.

9. Enter and confirm your custom identity keystore password, (for example, `welcome1`).

10. Under Trust, set the **Custom Trust Keystore** to `kss://system/trust` and click **Save**.

11. Enter `KSS` as the **Custom Trust Keystore Type**, and enter and confirm your custom trust keystore password, then click **Save**.

12. From the WLS Administration console, go to **Servers -> WC_Collaboration** and open the Configuration tab, and then the **SSL** subtab.

13. Enter the private key alias (for example, `collab_wls`), and set the private key password (for example, `welcome1`.

14. Click **Save** to have your entries.

15. On the Settings pane for the WebCenter Collaboration server (`WC_Collaboration`), open the Configuration tab and then the General subtab.

    The General Configuration pane opens.

16. Check **SSL Listen Port Enabled**.
17. Enter an **SSL Listen Port** number and click **Save**.
18. On the **Configuration** tab, open the **SSL** subtab, and then expand the Advanced options at the bottom of the page.
19. Check that the **Two Way Client Cert Behavior** option is set to `Client Certs Not Requested` and click **Save**.
20. Open the Control tab.

    The Control Settings pane opens.

21. Click **Restart SSL**.

Configuring and Securing the SSL Connection for Discussions

To configure the SSL connection for Discussions:

1. Restart the WebCenter Collaboration server (`WC_Collaboration`) server and open the SSL collaboration URL: `https://host:port/owc_discussions`.

   The certificate should be generated when you access the URL, and stored in your browser.

2. Download and store the certificate in .PEM or .CRT format.
3. Import the certificate into `cacerts` in `JDK_HOME`, using the following command:

   ```
   keytool -importcert -alias collab_cert -file /filepath/sslcertificate/
   collabcert.crt -keystore..../oracle_common/jdk/jre/lib/security/cacerts
   ```

4. Enter the password `changeit` when asked, then enter `YES`.
5. Register the `https://jive` URL in Oracle Enterprise Manager for Announcements and Discussions.
6. Restart the `WC_Portal` managed server.
7. Test announcements and discussions.

Securing the WebCenter Portal Connection to Portlet Producers with SSL

Securing the connection to WSRP with SSL is described in the following sections:

- Creating the Custom Keystores for Portlet Producers
- Configuring the Identity and Trust Keystores for Portlet Producers
- Configuring the SSL Connection for Portlet Producers
- Registering the SSL-enabled WSRP Producer and Running the Portlets

Creating the Custom Keystores for Portlet Producers

The following steps are required to configure WebCenter Portlet with SSL using the KSS keystore.

1. Connect to WebLogic Server using the WLST console:

   ```
   connect('weblogic','password','host:port')
   ```

2. Get the OPSS service reference:

   ```
   svc = getOpssService(name='KeyStoreService')
   ```

3. Create a new keystore:

   > **Note:**
   >
   > Create a keystore in the system stripe and the permission must be false.

   ```
   svc.createKeyStore(appStripe='system', name='portlet_wls',
   password='password', permission=false)
   ```

   where:

   - *svc* = the service command object obtained through a call to `getOpssService()`
   - *appStripe* = the name of the stripe in which the keystore is created
   - *name* = the name of the keystore
   - *password* = password of the keystore
   - *permission* = false if protected by both permission and password (true if keystore is protected by permission only)

4. Generate keypair:

   ```
   svc.generateKeyPair(appStripe='system', name='portlet_wls',
   password='password', dn='cn=customidentity,dc=example,dc=com',
   keysize='2048', alias='portlet_wls', keypassword='password')
   ```

   where:

   - *svc* = the service command object obtained through a call to `getOpssService()`
   - *appStripe* = the name of the stripe containing the keystore
   - *name* = the name of the keystore where the key pair is generated
   - *password* = password of the keystore
   - *dn* = the distinguished name of the certificate wrapping the key pair
   - *keysize* = the key size
   - *alias* = the alias of the key pair entry
   - *keypassword* = the key password

5. Optionally, list the keystores and aliases inside the keystore.

This will list the `system/portlet_wls`:

```
svc.listKeyStores(appStripe='*')
```

- *svc* = the service command object obtained through a call to `getOpssService()`
- *appStripe* = the name of the stripe whose keystores are listed

This will list the alias `portlet_wls`:

```
svc.listKeyStoreAliases(appStripe="system",name="portlet_wls",
password="password", type="*")
```

- *svc* = the service command object obtained through a call to `getOpssService()`
- *appStripe* = the name of the stripe containing the keystore
- *name* = the name of the keystore
- *password* = password of the keystore
- *type* = the type of entry for which aliases are listed. Valid values are `'Certificate'`, `'TrustedCertificate'`, `'SecretKey'` or `'*'`

6. Run `syncKeyStores`:

```
syncKeyStores(appStripe='system', keystoreFormat='KSS')
```

Configuring the Identity and Trust Keystores for Portlet Producers

The next step is to configure the Custom Identity and Trust Keystores for the WebCenter Portlet server (for example, `WC_Portlet`).

For an overview of on how to configure the Identity and Trust keystores, see Securing the Browser Connection to WebCenter Portal using SSL.

To configure the identity and trust keystores for the Portlet server:

1. Log in to the WebLogic Server Administration Console.

   For information on logging into the WebLogic Server Administration Console, see 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 WebCenter 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. Click **Change**.

6. For Keystores, select **Custom Identity and Custom Trust**, and click **Save**

7. Under **Identity**, enter the path and filename of the Custom Identity Keystore you created in `kss://system/portlet_wls` (Creating the Custom Keystores for Portlet Producers).

8. Enter `KSS` as the **Custom Identity Keystore Type**.

9. Enter and confirm your custom identity keystore password, (for example, `welcome1`).
10. Under Trust, set the **Custom Trust Keystore** to `kss://system/trust` and click **Save**.
11. Enter `KSS` as the **Custom Trust Keystore Type**, and enter and confirm your custom trust keystore password, then click **Save**.
12. Open the **SSL** tab.
13. Enter the private key alias (for example, `portlet_wls`), and set the private key password (for example, `welcome1`).
14. Click **Save** to save your entries.

Configuring the SSL Connection for Portlet Producers

To configure SSL, see Overview of Configuring SSL in WebLogic server in *Administering Security for Oracle WebLogic Server* guide.

To configure the SSL connection for Portlet Server:

1. On the **Settings** pane for the WebCenter Portlet server (`WC_Portlet`), open the **Configuration** tab and then the **General** subtab.
2. Select **SSL Listen Port Enabled**.
3. Enter an SSL listen port number.
4. Click **Save**.
5. Select **Configuration** > **SSL**, and then open the Advanced options at the bottom of the page.
6. Check that the **Two Way Client Cert Behavior** option is set to `Client Certs Not Requested` .
7. Click **Save**.
8. Open the **Control** tab.

   The Control Settings pane opens.
9. Click **Restart SSL**.
10. Restart the Portlet Server (`WC_Portlet`) and open the SSL WSRP Portlet URL: `https://host:port/<context-root>/portlets/wsrp2?WSDL`.
11. Accept the certificate for the session and WSDL will get loaded.

Registering the SSL-enabled WSRP Producer and Running the Portlets

Configure the `WC_Portal` managed server to register portlets with WebCenter Portal. This also uses the certificates in `JAVA_HOME` trust store (`/jdk/jre/lib/security/cacerts`).

To register the SSL-enabled WSRP producer and run the portlets:

1. When you accessed the SSL WSRP Portlet URL (`https://host:port/<context-root>/portlets/wsrp2?WSDL`), the certificate was generated and stored in your browser.
2. Download the certificate and save it in `.PEM` or `.crt` 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 der2pem in *Command Reference for Oracle WebLogic Server*.
3. Import the certificate into the `cacerts` file in the `/jdk/jre/lib/security` using the following keytool command:

   ```
   keytool -importcert -alias portlet_cert -file portlet_pem -keystore cacerts
   ```

   Where:

   - *`portlet_cert`* is the portlet certificate alias
   - *`portlet_pem`* is the portlet certificate file (for example, `portlet_cert.pem`)
4. Restart `WC_Portal`.
5. Register the SSL enabled portlet URL — 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')
   ```
6. Navigate to the HTTP or HTTPS WebCenter Portal URL.
7. Create a page and go to the Portlets link.
8. Go to the registered WSRP producer.
9. Add the portlet to the page.
10. Go to the view mode of the page and check that the WSRP portlet renders correctly.

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 Configuring the Identity Store for the steps to do this for the identity store.

If the `CA` is unknown to the Oracle WebLogic server, complete this additional step described in the following subsection:

- Exporting the OID Certificate Authority (CA)

Exporting the OID Certificate Authority (CA)

The following topics describe how to secure the WebCenter Portal connection to OID:

1. Enabling the SSL in OID
2. Importing the OID Certificate
3. Establishing the SSL Connections

Enabling the SSL in OID

This topic describes how to enable the SSL in OID.

> **Note:**
>
> OID should be configured in the `server auth` mode.

1. Create an Oracle wallet by running the following commands:

```
<OID_INSTALL_LOC>/oracle_common/bin/orapki wallet create -wallet
<wallet_location>/OID_Wallet -auto_login
```

> **Note:**
>
> Enter the password, when prompted.

where,

- `<OID_INSTALL_LOC>` is the location where the OID is installed.

  `<wallet_location>` is the location where you want the new wallet named `OID_Wallet` to be created. If you do not specify the wallet location, the new wallet is created in the current directory, where the command is executed.

2. Add certificates to an Oracle wallet by running the following commands:

```
<OID_INSTALL_LOC>/oracle_common/bin/orapki wallet add -wallet -
wallet <wallet_location>/OID_Wallet -dn cn=<Domain name> -keysize
2048 -self_signed -sign_alg sha1 -validity 1000
```

Where,

- `<OID_INSTALL_LOC>` is the OID install location.
- `<wallet_location>` is the wallet location.
- `cn` is the domain name where OID server is installed. You can find the domain name from `/etc/hosts` file.

  For example: `cn=<Domain name>`.

- `-sign_alg` is signature algorithm. MD5 is the default value of signature algorithm.

  The recent versions of `JDK`, which is `JDK8` does not support the MD5 algorithm, you need to give **`sha1`** or **`sha2`** for the signature algorithm. For example: `sha1` .

- `-self_signed` is a self signed certificate.

  You can also get the certificate trusted by CA and import it accordingly. For more information, see Configuring Secure Sockets Layer (SSL).

3. Configure the SSL parameters in OID by running the following commands:

```
ldapmodify -h OID_host -p OID_port -D cn=OID_admin -w password
dn:cn=oid1,cn=osdldapd,cn=subconfigsubentry
changetype: modify
replace: orclsslauthentication
orclsslauthentication: 32
-
replace: orclsslwalleturl
orclsslwalleturl: file://<wallet_location>/OID_wallet
```

4. Restart the OID server.
5. Verify that the SSL connections are created successfully by running the following commands:

```
./ldapbind -h OID_host -p OID_port -U 2 -W file://<wallet_location>/
OID_Wallet  -P password
```

   where,

   - `<wallet_location>/OID_Wallet` is wallet location.

6. Export the certificate by running the following command:

```
<OID_INSTALL_LOC>/oracle_common/bin/orapki wallet export -wallet /
<wallet_location>/OID_Wallet -dn "cn=<Domain name>" -cert oid_trust.cer
```

   where,

   - `<OID_install_LOC>/<wallet_location>/OID_Wallet` is the location of the wallet.
   - `oid_trust.cer` is the certificate. By default, the wallet certificate is created in the current directory where the command is executed. If you specify the path, wallet certificate is created in the specified location, for example: `/OID_Install_LOC/oid_cert_trust.cer`.

Importing the OID Certificate

This topic describes how to import the OID certificate to the WebLogic Server Trust Store of WebCenter.

> **Note:**
>
> The procedure has to be performed on your WebLogic domain, where the WebCenter Portal server is installed.

1. Import the certificate to the Oracle WebLogic Server Trust Store of the WebCenter Portal using the following command:

   ```
   keytool -importcert -v -trustcacerts -alias oid_server_trust -file
   oid_trust.cer -keystore cacerts -storepass changeit
   ```

   > **Note:**
   >
   > The `cacerts` path can be retrieved as follows:
   >
   > a. Log in to the WebLogic console, navigate to **Servers** and click `WC_Portal` server.
   >
   > b. Click **Configurations**, then click the **Keystores** subtab.
   >
   > c. Verify the path mentioned in the Java Standard Trust Keystore.
   >
   > > **Note:**
   > >
   > > The path mentioned in the Java Standard Trust Keystore is your `cacert` path.

2. Configure the OID with Oracle WebLogic Server.

   For more information, see Configuring the Oracle Internet Directory Authenticator.

   > **Note:**
   >
   > When entering the Provider Specific information, ensure to specify an SSL host and port and to select the **SSL Enabled** check box.

Establishing the SSL Connections

This topic describes how to Establish the SSL connections between the identity store and LDAP server.

> **Note:**
>
> The procedure has to be performed on your WebLogic domain, where the WebCenter Portal server is installed.

1. Set up your environment using the following script:

```
setenv WL_HOME <WCP_INSTALL_LOCATION>/wlserver
setenv ORACLE_HOME <WCP_ORACLE_HOME>
cd $WL_HOME/server/bin
./setWLSEnv.sh
cd $ORACLE_HOME/oracle_common/bin
```

2. Create the keystore using the following script:

```
libovdconfig.sh -host wls_host -port wls_adminserver_port -userName
wls_user_name -domainPath full_path_domain_home -createKeystore
```

   - `host` is the Oracle WebLogic Server host
   - `port` is the Oracle WebLogic Server Admin Server port
   - `username` is the Oracle WebLogic Server admin user name
   - `domainPath` is the complete path to the domain home

> **Note:**
>
> The keystore is created in the following location `-keystore $DOMAIN_HOME/config/fmwconfig/ovd/default/keystores/adapters.jks`

3. Import the certificate to the keystore using the `keytool` command. The syntax is as follows, for a keystore named `adapters.jks`.

   Ensure that you have exported the previously generated OID. For more information, see Enabling the SSL in OID.

> **Note:**
>
> The keystore `adapters.jks`is created in Step 2.

```
$JAVA_HOME/bin/keytool -importcert
-keystore $DOMAIN_HOME/config/fmwconfig/ovd/default/keystores/
adapters.jks
-storepass keystore_password_used_in_libovdconfig.sh
-alias alias_name
-file full_path_to_LDAPCert_file
-noprompt
```

4. Restart the Oracle WebLogic Server and the managed servers.
5. Access the WebCenter Portal and log in as any OID user. You should be able to login successfully.

> **Note:**
>
> if you receive host name verification exception, then set the following parameter:
>
> ```
> -Dweblogic.security.SSL.ignoreHostnameVerification=true
> ```

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 der2pem in *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 Registering Mail Servers.
8. Restart WebCenter Portal.
9. Log into WebCenter Portal and provide your mail credentials.

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 Reassociating the Identity Store with an External LDAP Server.

To secure the WebCenter Portal connection to an external BPEL server with SSL:

1. Install and configure Oracle SOA 12*c*.

   See Installing Oracle SOA Suite Quick Start for Developers in *Installing SOA Suite and Business Process Management Suite Quick Start for Developers*.

2. From WebCenter, create a connection to SOA in WebCenter, by running the following commands:

   ```
   createBPELConnection('webcenter','WebCenter-Worklist'
   setSpacesWorkflowConnectionName('webcenter', 'WebCenter-Worklist',
   'SOA_host:port','oracle/wss10_saml_token_client_policy')
   ```

3. From WebCenter, enable SSL.

   Follow the steps in Securing the Browser Connection to WebCenter Portal using SSL.

4. From SOA, enable SSL.

   Follow the steps in Securing the Browser Connection to WebCenter Portal using SSL, but instead of `webcenter_wls`, you will use `soa_wls` and instead of `webcenteridentity`, you will use `soaidentity`.

5. Configure the keystores for WebCenter Portal and SOA.

   See Creating the WebCenter Portal Domain Keystore and Creating the SOA Domain Keystore.

6. Wire WebCenter WebLogic server and SOA WebLogic server to the same OID.

7. From WebCenter, import the SOA public and CA certificate to the WebCenter Trust store:

   ```
   keytool -importcert -trustcacerts -alias soa_cert -file /filepath/certificate/
   bpel.cer -keystore /filepath/cacerts -storepass changeit

   keytool -importcert -trustcacerts -alias soa_trust -file /filepath/certificate/
   democabpel.cer -keystore /filepath/cacerts -storepass changeit
   ```

8. From SOA, import the WebCenter public and CA certificate to the SOA Trust Store:

   ```
   keytool -importcert -trustcacerts -alias webcenter_cert -file /filepath/
   certificate/webcenter.cer -keystore /filepath/cacerts -storepass changeit
   ```

```
keytool -importcert -trustcacerts -alias webcenter_trust -file /filepath/
certificate/democaprod.cer -keystore /filepath/cacerts -storepass changeit
```

9. From WebCenter, change the SOA connection details to use the SOA HTTPS host and port in Oracle Enterprise Manager.
10. Add `-Dweblogic.security.SSL.ignoreHostnameVerification=true` as `EXTRA_JAVA_PROPERTIES` in `setDomainEnv.sh` for Webcenter.
11. Restart the `WC_Portal` server and the SOA managed server.

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 the connection with SSL is described in the following sections:

- Configuring Content Server SSL for Socket SSL
- Creating Content Server SSL Connection in WebCenter Portal

Configuring Content Server SSL for Socket SSL

To configure Content Server socket SSL, see Configuring SSL in *Developing with Oracle WebCenter Content*.

Note the following parameters at the end of the configuration:

- `serverPort`
- `keystoreLocation` - *client_keystore*
- `keystorePassword` - *idcidc*
- `privateKeyAlias` - *SecureClient*
- `privateKeyPassword` - *idcidc*

Creating Content Server SSL Connection in WebCenter Portal

Creating Socket SSL Connection

Create Content Server Connection in WebCenter Portal

- To create Content Server connection, do the following:

```
createContentServerConnection (appName='webcenter', name='dev-ucm',
socketType='socketssl', serverHost='100.111.149.197',
serverPort='3333', isPrimary='true', keystoreLocation='/scratch/
patch-jars/ucm_keystore/client_keystore',
keystorePassword='idcidc', privateKeyAlias='SecureClient',
privateKeyPassword='idcidc')
setContentServerProperties(appName='webcenter',
```

```
portalServerIdentifier='/WCP01', securityGroup='WCP01',
adminUserName='weblogic')
```

> **Note:**
>
> The `serverPort` must be SSL Port. For keystore configuration, see Configuring Content Server SSL for Socket SSL. The values that you have noted in the previous step like keystore and aliases will be used when you will create content server connection using WLST command.

For createContentServerConnection details, see WebCenter Portal Custom WLST Commands in *WebCenter WLST Command Reference* and Managing Connections to Oracle WebCenter Content Server.

- Restart the Portal server.
- Now access the WebCenter Content RUI and test the documented use cases.

Creating JAX-WS SSL Connection

To import the WebCenter Content SSL Certificate to WebCenter Portal Keystore (cacerts):

1. Enter the Content Server URL: `https://host:port/cs` in the firefox browser. The certificate should be generated when you access the URL, and stored in your browser.
2. Download and store the certificate in .PEM or .CRT format.
3. Import the certificate into cacerts in `JDK_HOME`, using the following command:

```
keytool -importcert -alias collab_cert -file /filepath/sslcertificate/
contentcert.crt -keystore..../oracle_common/jdk/jre/lib/security/cacerts
```

4. Enter the password `changeit` when asked, then enter `YES`.

Creating Content Server Connection in WebCenter Portal

- To create Content Server connection, do the following:

```
createContentServerConnection(appName='webcenter', name='dev-ucm',
socketType='jaxws', url='https://<ucm_host>:<sslPort>/idcnativews',
clientSecurityPolicy='oracle/wss10_saml_token_client_policy',
adminUsername='weblogic',isPrimary='true')
setContentServerProperties(appName='webcenter', portalServerIdentifier='/
WCP01', securityGroup='WCP01', adminUserName='weblogic')
```

> **Note:**
>
> The URL must be https URL of WebCenter Content.

- Restart the Portal server.
- Now access the WebCenter Content RUI and test the documented use cases.

For createContentServerConnection details, see WebCenter Portal Custom WLST Commands and Managing Connections to Oracle WebCenter Content Server.

28

Configuring Web Services Security

Configure Web Services Security (WS-Security) for WebCenter Portal and related services and components.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

WS-Security, using an OPSS Key Store Service (KSS) keystore, provides a mechanism for retrieving and managing the security credentials of a WebCenter Portal application and ancillary applications and components across one or more domains. The KSS keystore provides information about available public and private keys that can be used for authentication and data integrity.

The following topics show how to configure a typical topology with WS-Security (where the WebCenter Portal application and the WSRP producers share the same domain, but the BPEL server is in an external SOA domain), and how to extend that configuration for more complex environments (where, for example, a second discussions server is in a separate domain, a BPEL server is in a separate SOA domain, and one WSRP producer is in an external portlet domain):

> **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 Understanding Administrative Operations, Roles, and Tools.

Topics:

- Configuring WS-Security for a Typical Topology
- Configuring WS-Security for Multiple Domains
- Securing WebCenter Portal for Applications Consuming WebCenter Portal Client API with WS-Security

Configuring WS-Security for a Typical Topology

This section describes how to configure WS-Security for a topology where the WebCenter Portal application, WSRP producers, and discussions server share the same domain, but the BPEL (SOA) server is in an external domain.

Typical Topology

- Domain 1 : WebCenter Portal , Discussions, Portlet Producers
- Domain 2 : SOA

The steps to configure WS-Security for a typical two-domain topology are described in the following topics:

- Creating the WebCenter Portal Domain Keystore
- Creating the SOA Domain Keystore
- Configuring the Discussions Server

Creating the WebCenter Portal Domain Keystore

This section describes how to use the OPSS Keystore Service (KSS) to create the WebCenter Portal keystore and keys. 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 keystore. After creating the keystore, the security credentials of WebCenter Portal, discussions server, BPEL servers, and WSRP producers can be retrieved and managed using the KSS. For more information about the OPSS Keystore Service, see Managing Keys and Certificates with the Keystore Service in *Securing Applications with Oracle Platform Security Services*.

To create the WebCenter Portal domain keystore:

1. Run the following WLST commands:

   ```
   svc = getOpssService(name='KeyStoreService')
   ```

2. Create the keystore using the following WLST command:

   ```
   svc.createKeyStore(appStripe='appStripe', name='producer',
   password='password', permission=true/false))
   ```

   Where:

   - *appstripe* — The keystore stripe name. Keys and certificates created in the keystore reside in an application stripe or product, and each stripe in a domain is uniquely named
   - *name* — Keystore name
   - *password* — Keystore password
   - *permission* — false if protected by both permission and password (true if keystore is protected by permission only)

For example:

```
svc.createKeyStore(appStripe='WCPortalStripe', name='producer',
password='welcome1', permission=true))
```

3. Generate the key pair for this newly created keystore:

```
svc.generateKeyPair(appStripe='appstripe', name='name',
password='password', dn='CN=Producer, OU=Producer, O=MyOrganization,
L=MyTown, ST=MyState, C=US', keysize='2048', alias='producer',
keypassword='keypassword')
```

Where:

- *appstripe* — The keystore stripe name. Keys and certificates created in the keystore reside in an application stripe or product, and each stripe in a domain is uniquely named
- *name* — Keystore name
- *password* — Keystore password
- *dn* — Domain name (for example, `dn='CN=webcenter_certificate'`)
- *alias* — Public Key Alias
- *keypassword* — Password for new public key

For example:

```
svc.generateKeyPair(appStripe='WCPortalStripe', name='producer',
password='welcome1', dn='CN=Producer, OU=Producer, O=MyOrganization,
L=MyTown, ST=MyState, C=US', keysize='2048', alias='producer',
keypassword='welcome1')
```

4. Export the producer certificate (which will be used by the consumer):

```
svc.exportKeyStoreCertificate(appStripe='appstripe', name='name',
password='password', alias='alias',
type='TrustedCertificate',filepath='filepath')
```

Where:

- *appstripe* — The keystore stripe name. Keys and certificates created in the keystore reside in an application stripe or product, and each stripe in a domain is uniquely named
- *name* — Keystore name
- *password* — Keystore password
- *alias* — Public Key Alias
- *keypassword* — Password for new public key
- *filepath* — Certificate path

For example:

```
svc.exportKeyStoreCertificate(appStripe='WCPortalStripe',
name='producer', password='welcome1', alias='producer',
type='TrustedCertificate',filepath='/scratch/certificate/
webcenter.cer')
```

Creating the SOA Domain Keystore

This section describes how to create a SOA domain keystore and keys using an OPSS keystore (KSS). For syntax and reference information about the KSS commands, see OPSS Keystore Service Commands in *Oracle Fusion Middleware Infrastructure Security WLST Command Reference*.

To create the SOA domain keystore:

1. Using the following WLST command, get an OPSS service command object:

   ```
   svc = getOpssService(name='KeyStoreService')
   ```

2. Create the keystore:

   ```
   svc.createKeyStore(appStripe='appStripe', name='name',
   password='password', permission=true/false))
   ```

   Where:

   - *appstripe* — The keystore stripe name. Keys and certificates created in the keystore reside in an application stripe or product, and each stripe in a domain is uniquely named
   - *name* — Keystore name
   - *password* — Keystore password
   - *permission* — false if protected by both permission and password (true if keystore is protected by permission only)

   For example:

   ```
   svc.createKeyStore(appStripe='SOAStripe', name='bpel',
   password='welcome1', permission=true))
   ```

3. Generate key pair for the newly created keystore:

   ```
   svc.generateKeyPair(appStripe='appstripe', name='name',
   password='password', dn='CN=Producer, OU=Producer,
   O=MyOrganization, L=MyTown, ST=MyState, C=US', keysize='2048',
   alias='bpel', keypassword='keypassword')
   ```

   Where:

   - *appstripe* — The keystore stripe name. Keys and certificates created in the keystore reside in an application stripe or product, and each stripe in a domain is uniquely named

- *name* — Keystore name
- *password* — Keystore password
- *dn* — Domain name (for example, `dn='CN=webcenter_certificate'`)
- *alias* — Public Key Alias
- *keypassword* — Password for new public key

For example:

```
svc.generateKeyPair(appStripe='SOAStripe', name='bpel',
password='welcome1', dn='CN=BPEL, OU=Consumer, O=MyOrganization,
L=MyTown, ST=MyState, C=US', keysize='2048', alias='bpel',
keypassword='welcome1')
```

4. Import the certificate exported by the producer:

```
svc.importKeyStoreCertificate(appStripe='appStripe', name='name',
password='password', alias='webcenter_spaces_ws',
keypassword='keypassword', filepath='filepath',type='TrustedCertificate')
```

Where:

- *appstripe* — The keystore stripe name. Keys and certificates created in the keystore reside in an application stripe or product, and each stripe in a domain is uniquely named
- *name* — Keystore name
- *password* — Keystore password
- *keypassword* — Password for new public key
- *filepath* — Certificate path

> **Note:**
>
> The alias for the `importKeyStoreCertificate` command must always be set to `webcenter_spaces_ws`. Do not attempt to change this alias or the SOA usecases will fail.

For example:

```
svc.importKeyStoreCertificate(appStripe='SOAStripe', name='bpel',
password='welcome1', alias='webcenter_spaces_ws', keypassword='welcome1',
filepath='/scratch/certificate/webcenter.cer',type='TrustedCertificate')
```

5. Export the public certificate that will be imported by the producer:

```
svc.exportKeyStoreCertificate(appStripe='appstripe', name='name',
password='password', alias='alias',
filepath='filepath',type='TrustedCertificate')
```

Where:

- *appstripe* — The keystore stripe name. Keys and certificates created in the keystore reside in an application stripe or product, and each stripe in a domain is uniquely named
- *name* — Keystore name
- *password* — Keystore password
- *alias* — Public Key Alias
- *filepath* — Certificate path

For example:

```
svc.exportKeyStoreCertificate(appStripe='SOAStripe', name='bpel',
password='welcome1', alias='bpel', filepath='/scratch/certificate/
bpel.cer',type='TrustedCertificate')
```

6. Register the newly created stripe in SOA domain:

```
configureWSMKeystore('/WLS/base_domain','KSS', 'kss://appstripe/
bpel', signAlias='bpel', cryptAlias='bpel',
signAliasPassword='signAliasPassword',cryptAliasPassword='cryptAlias
Password')
```

Where:

- *WLS/base_domain* — The domain name and should follow the format: *WLS/<domainName>*
- *kss://appstripe/bpel* — The KSS keystore name
- *cryptAlias* — The public key alias
- *appstripe* — The keystore stripe name. Keys and certificates created in the keystore reside in an application stripe or product, and each stripe in a domain is uniquely named
- *signAliasPassword* — The password for the public key
- *cryptAlias* — The public key alias
- *cryptAliasPassword* — The password for the public key

For example:

```
configureWSMKeystore('/WLS/base_domain','KSS', 'kss://SOAStripe/
bpel', signAlias='bpel', cryptAlias='bpel',
signAliasPassword='signAliasPassword',cryptAliasPassword='cryptAlias
Password')
```

7. Grant keystore permission to newly created `bpel` stripe in the SOA domain:

```
grantPermission(permClass="oracle.security.jps.service.keystore.KeyS
toreAccessPermission",
permTarget="stripeName=SOAStripe,keystoreName=bpel,alias=*",
permActions="read")
```

8. Import the consumer certificate to WebCenter:

```
svc.importKeyStoreCertificate(appStripe='appstripe', name='name',
password='password', alias='bpel', keypassword='keypassword',
filepath='filepath', type='TrustedCertificate')
```

Where:

- *appstripe* — The keystore stripe name. Keys and certificates created in the keystore reside in an application stripe or product, and each stripe in a domain is uniquely named
- *name* — Keystore name
- *password* — Keystore password
- *alias* — Public Key Alias
- *keypassword* — Password for new key
- *filepath* — Certificate path

For example:

```
svc.importKeyStoreCertificate(appStripe='WCPortalStripe',
name='producer', password='welcome1', alias='bpel',
keypassword='welcome1', filepath='/scratch/certificate/bpel.cer',
type='TrustedCertificate')
```

9. Register the producer stripe:

```
configureWSMKeystore('/WLS/wc_domain','KSS', 'kss://appstripe/producer',
signAlias='producer', signAliasPassword='signAliasPassword',
cryptAlias='cryptAlias', cryptAliasPassword='cryptAliasPassword')
```

Where:

- *wc_domain* — The WebCenter Portal domain
- *signAliasPassword* — The password for the public key
- *cryptAlias* — The public key alias
- *cryptAliasPassword* — The password for the public key

For example:

```
configureWSMKeystore('/WLS/WLS_SOAWC','KSS', 'kss://WCPortalStripe/
producer', signAlias='producer', signAliasPassword='welcome1',
cryptAlias='producer', cryptAliasPassword='welcome1')
```

10. Grant Keystore Permission for the newly created stripe:

```
grantPermission(permClass="oracle.security.jps.service.keystore.KeyStoreAc
cessPermission",
permTarget="stripeName=WCPortalStripe,keystoreName=producer,alias=*",
permActions="read")
```

Configuring the Discussions Server

If the discussions server for your topology is in the same domain as the `WC_Portal` server and is not being used in a production environment, then no extra keystore configuration is needed since the keystore configured for the WebCenter Portal domain is used for the discussions server 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 topics:

- Attaching Security Policies for WebCenter Portal and Discussions Web Service Endpoints
- Securing the Discussions End Points
- Configuring the Discussions Server Connection Settings

> **Note:**
>
> Discussions-specific web services messages sent by WebCenter Portal 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 Configuring SSL .

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 Securing the Discussions End Points.

> **Note:**
>
> In a patched WebCenter Portal instance, you must determine the policy names before you patch, then verify that the policies are the same after an upgrade..

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_Portal` and `WC_Collaboration` managed servers are running.
2. 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')
```

3. Restart the `WC_Portal` and `WC_Collaboration` managed servers.

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 topics:

- Securing the Discussions Server End Points Using Fusion Middleware Control
- Securing the Discussions Server End Points Using WLST

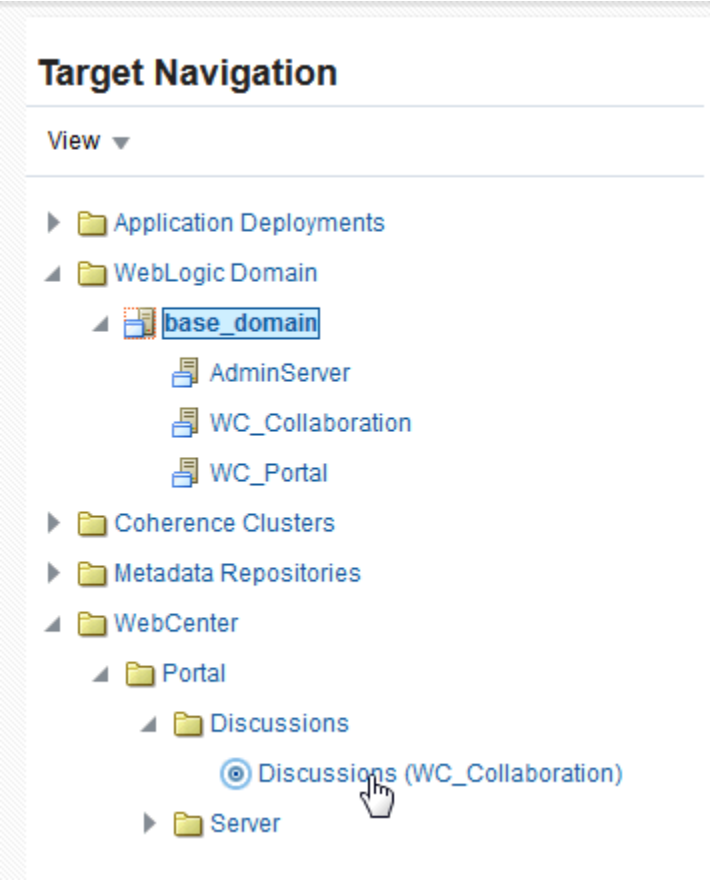
Securing the Discussions Server End Points Using Fusion Middleware Control

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 28-1).

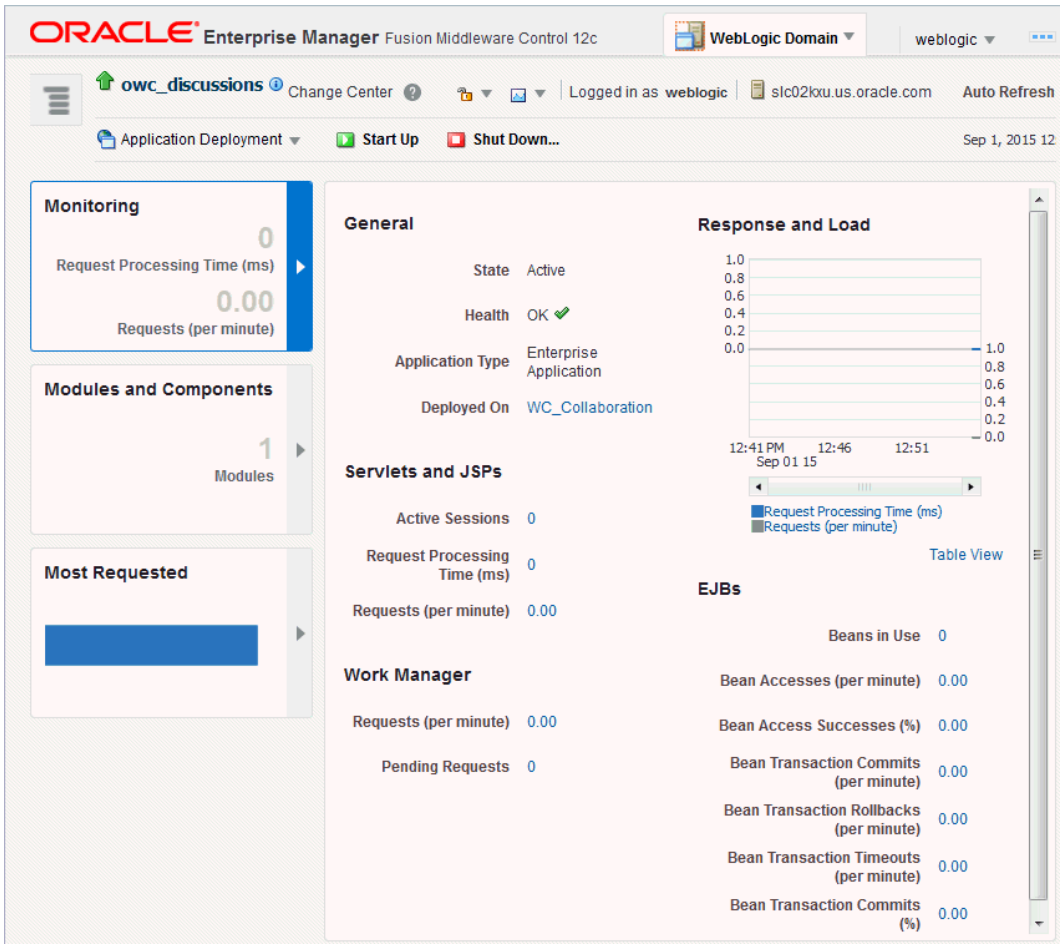
Figure 28-1 Discussions Home Page



2. Click the `owc_discussions` target.

   The home page for the `owc_discussions` application displays (see Figure 28-2).

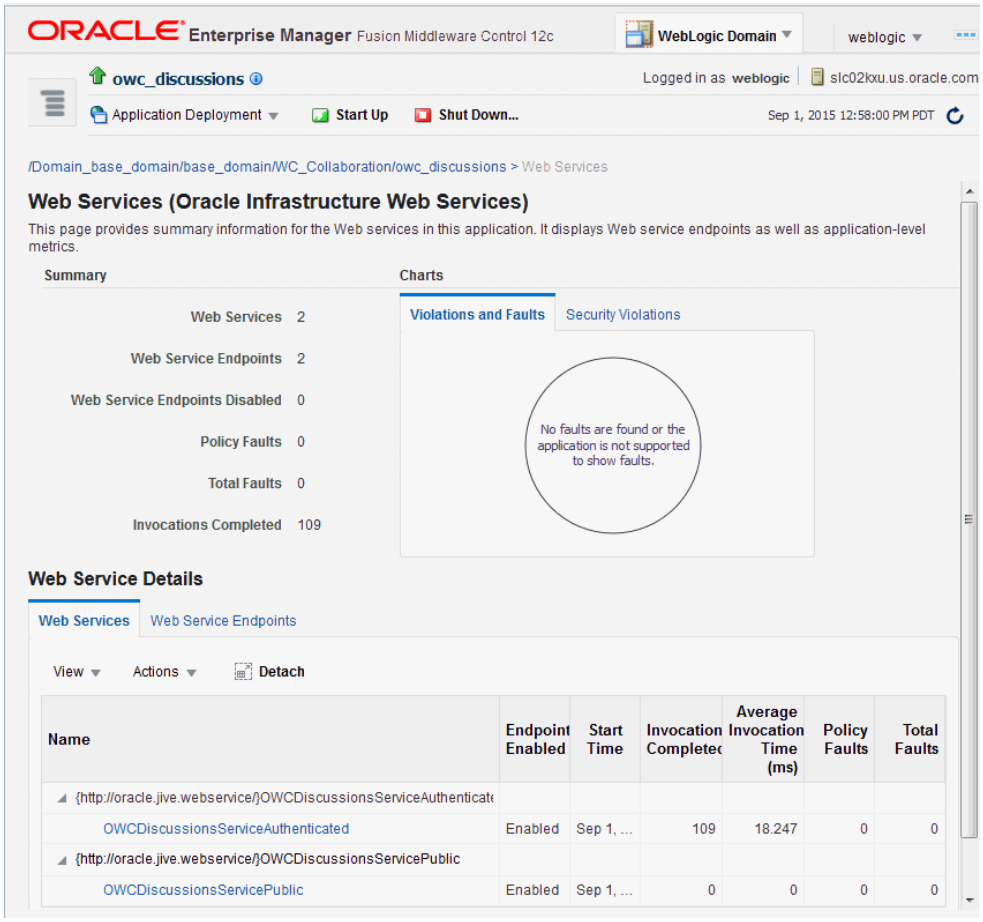
Figure 28-2 owc_discussions Home Page



3. From the Application Deployment menu, select **Web Services**.

   The Web Services page for the `owc_discussions` application displays (see Figure 28-3).

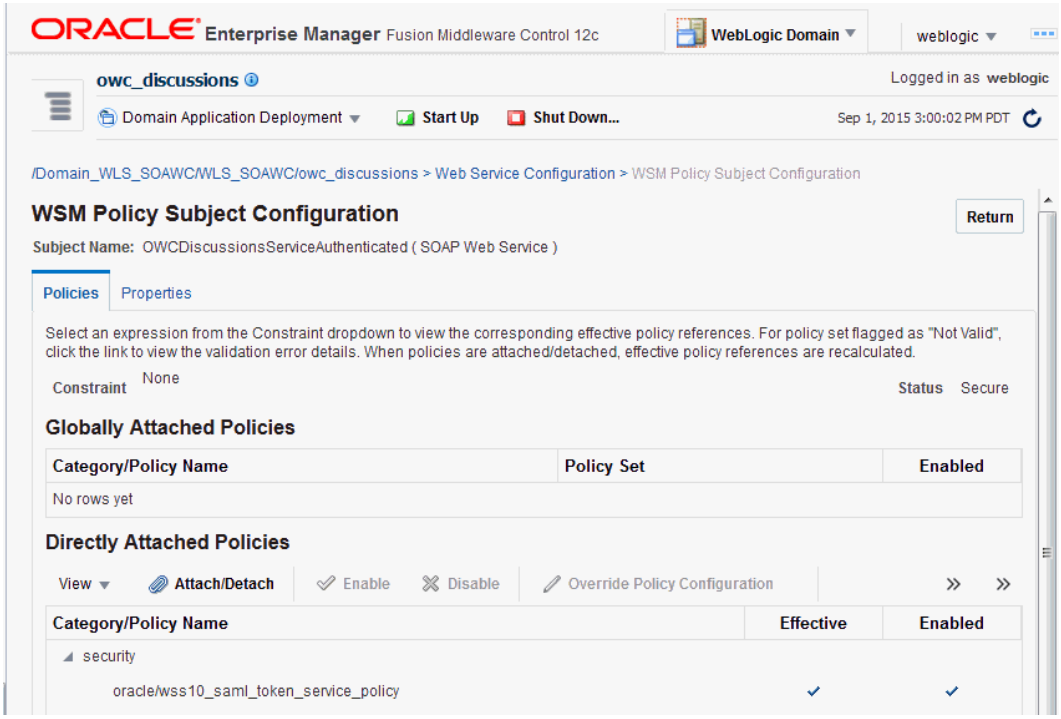
Figure 28-3 Web Services Page for owc_discussions



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 28-4).

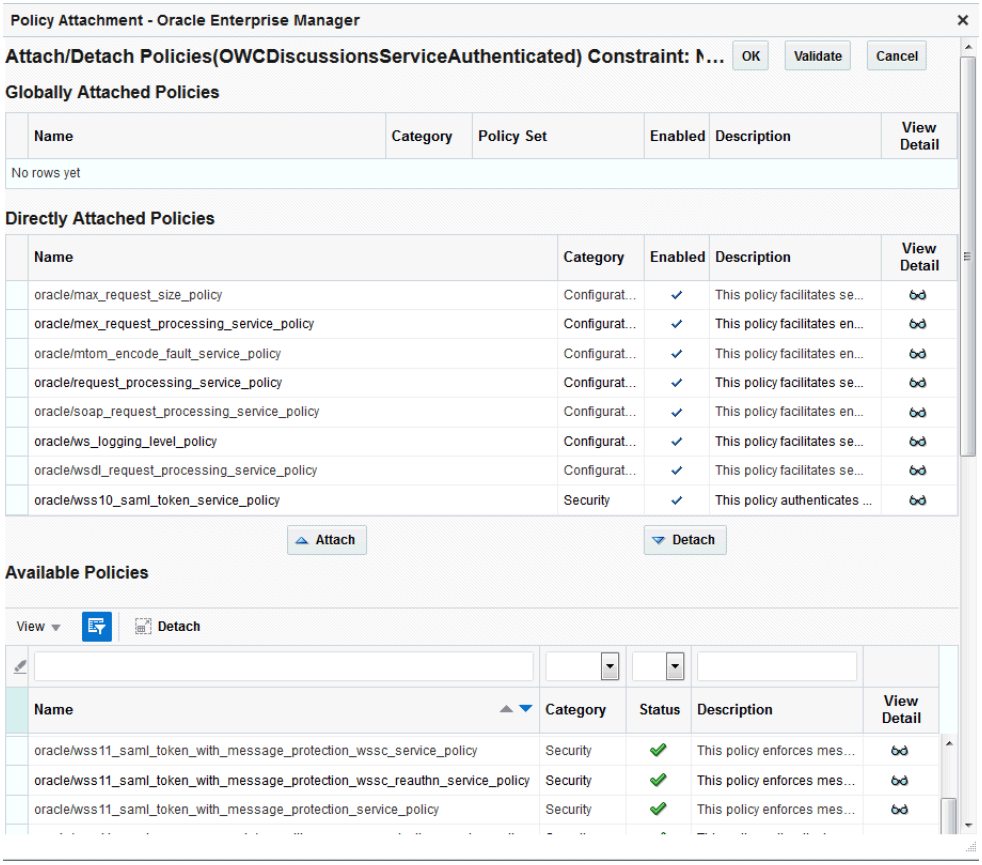
Figure 28-4 Web Service Endpoint Page



5. Click **Attach/Detach**.

   The Attach Policy page displays (see Figure 28-5).

Figure 28-5 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**.

Securing the Discussions Server End Points Using WLST

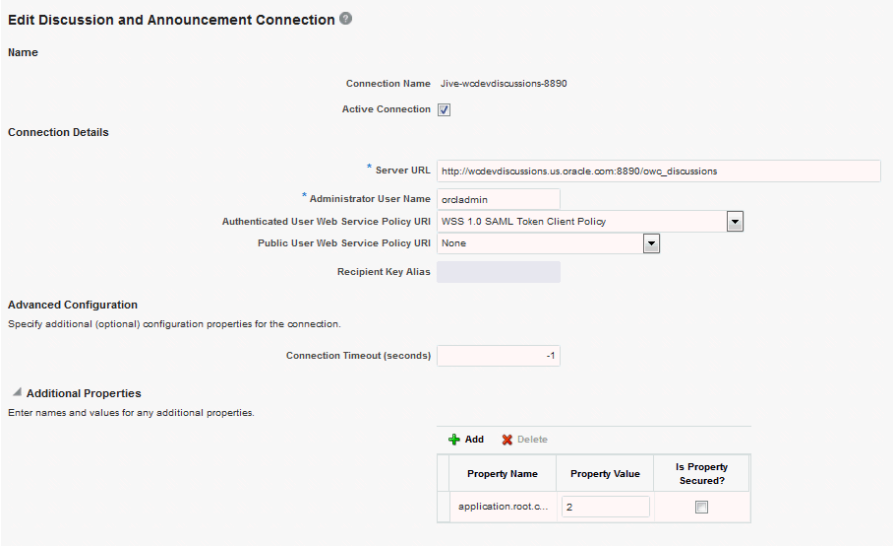
To secure the discussions server endpoints using WLST, detach the `wss10_saml_token_service_policy` and attach the `wss11_saml_token_with_message_protection_service_policy` using the following WLST commands:

```
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')
```

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 application, as described in Registering Discussions Servers. Figure 28-6 shows example connection detail settings for the Edit Discussions and Announcement Connection page.

Figure 28-6 Edit Discussions and Announcement Connection Page



Configuring WS-Security for Multiple Domains

This section describes how to extend the WS-security configuration for a typical topology for topologies where, for example, the WebCenter Portal application, BPEL (SOA) server, discussions server, and a WSRP producer server are each in their own domain.

Multiple Domain Topology

- Domain 1 : WebCenter Portal
- Domain 2 : SOA (BPEL) server
- Domain 3 : Discussions server
- Domain 4 : WSRP producers

The steps to configure WS-Security for a topology with multiple domains are described in the following topics:

- Setting Up the WebCenter Portal Domain Keystore
- Creating the SOA Domain Keystore
- Configuring an External Discussions Server
- Creating the External Portlet Domain Keystore

Setting Up the WebCenter Portal Domain Keystore

To create the WebCenter Portal domain keystore, follow the steps for a configuring WS-security for a typical topology as described in Creating the WebCenter Portal Domain Keystore. After creating the keystore, the security credentials of WebCenter Portal, discussions server, BPEL servers, and WSRP producers can be retrieved and managed using the KSS. For more information about the OPSS Keystore Service, see Managing Keys and Certificates with the Keystore Service in *Securing Applications with Oracle Platform Security Services*.

Creating the SOA Domain Keystore

Create the SOA domain keystore and keys using an OPSS keystore (KSS) as described in Creating the SOA Domain Keystore. For syntax and reference information about the KSS commands, see OPSS Keystore Service Commands in *Oracle Fusion Middleware Infrastructure Security WLST Command Reference*.

Configuring an External Discussions Server

If the discussions server is in a different domain than WebCenter Portal, 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:

- Securing the Discussions Service End Points
- Creating the Discussions Server Keystore
- Configuring the Discussions Server Connection Settings

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 Securing the Discussions End Points to secure the endpoints using either Fusion Middleware Control or WLST.

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. From the discussions server, run the following WLST command:

   ```
   svc = getOpssService(name='KeyStoreService')
   ```

2. Create the keystore:

   ```
   svc.createKeyStore(appStripe='appStripe', name='discussions',
   password='password', permission=true/false))
   ```

Where:

- *appstripe* — The keystore stripe name. Keys and certificates created in the keystore reside in an application stripe or product, and each stripe in a domain is uniquely named
- *name* — Keystore name
- *password* — Keystore password
- *permission* — false if protected by both permission and password (true if keystore is protected by permission only)

For example:

```
svc.createKeyStore(appStripe='dfstripe', name='discussions',
password='welcome1', permission=false)
```

3. Generate key pair for the newly created keystore:

```
svc.generateKeyPair(appStripe='appstripe', name='name',
password='password', dn='CN=Producer, OU=Producer, O=MyOrganization,
L=MyTown, ST=MyState, C=US', keysize='2048', alias='discussions',
keypassword='keypassword')
```

Where:

- *appstripe* — The keystore stripe name. Keys and certificates created in the keystore reside in an application stripe or product, and each stripe in a domain is uniquely named
- *name* — Keystore name (in this case “discussions”.
- *password* — Keystore password
- *dn* — Domain name
- *alias* — Public Key Alias (in this case “discussions”
- *keypassword* — Password for new public key

For example:

```
svc.generateKeyPair(appStripe='dfstripe', name='discussions',
password='welcome1', dn='CN=DISCUSSIONS, OU=Consumer, O=MyOrganization,
L=MyTown, ST=MyState, C=US', keysize='2048', alias='discussions',
keypassword='welcome1')
```

4. Import the certificate exported by the producer:

```
svc.importKeyStoreCertificate(appStripe='appStripe', name='name',
password='password', alias='webcenter_df_ws', keypassword='keypassword',
filepath='filepath',type='TrustedCertificate')
```

Where:

- *appstripe* — The keystore stripe name. Keys and certificates created in the keystore reside in an application stripe or product, and each stripe in a domain is uniquely named

- *name* — Keystore name (in this case “discussions”
- *password* — Keystore password
- *keypassword* — Password for new public key
- *filepath* — Certificate path

For example:

```
svc.importKeyStoreCertificate(appStripe='dfstripe',
name='discussions', password='welcome1', alias='webcenter_df_ws',
keypassword='welcome1', filepath='/scratch/certificate/
webcenter.cer',type='TrustedCertificate')
```

5. Export the public certificate that will be imported by the producer:

```
svc.exportKeyStoreCertificate(appStripe='appstripe', name='name',
password='password', alias='alias',
filepath='filepath',type='TrustedCertificate')
```

Where:

- *appstripe* — The keystore stripe name. Keys and certificates created in the keystore reside in an application stripe or product, and each stripe in a domain is uniquely named
- *name* — Keystore name
- *password* — Keystore password
- *alias* — Public Key Alias
- *filepath* — Certificate path

For example:

```
svc.exportKeyStoreCertificate(appStripe='dfstripe',
name='discussions', password='welcome1', alias='discussions',
filepath='/workplace/certificate/
discussions.cer',type='TrustedCertificate')
```

6. Register the newly created stripe:

```
configureWSMKeystore('/WLS/base_domain','KSS', 'kss://dfstripe/
discussions', signAlias='discussions', cryptAlias='discussions',
signAliasPassword='signAliasPassword',cryptAliasPassword='cryptAlias
Password')
```

Where:

- *signAliasPassword* — Password for the signature key alias
- *cryptAliasPassword* — Password for the encryption key alias

7. Grant keystore permission to newly created `discussions` stripe:

```
grantPermission(permClass="oracle.security.jps.service.keystore.KeyS
toreAccessPermission",
```

```
permTarget="stripeName=dfstripe,keystoreName=discussions,alias=*",
permActions="read")
```

8. Restart the managed servers and admin servers.
9. Import the discussion public certificate to WebCenter:

```
svc.importKeyStoreCertificate(appStripe='appstripe', name='name',
password='password', alias='discussions', keypassword='keypassword',
filepath='filepath', type='TrustedCertificate')
```

Where:

- *appstripe* — The keystore stripe name. Keys and certificates created in the keystore reside in an application stripe or product, and each stripe in a domain is uniquely named
- *name* — Keystore name
- *password* — Keystore password
- *alias* — Public Key Alias
- *keypassword* — Password for new key
- *filepath* — Certificate path

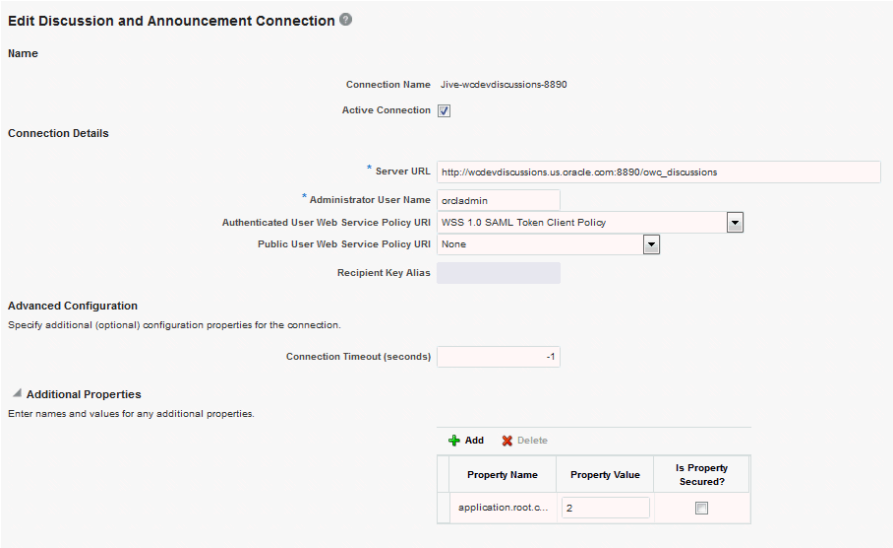
For example:

```
svc.importKeyStoreCertificate(appStripe='WCPortalStripe',
name='producer', password='welcome1', alias='discussions',
keypassword='welcome1', filepath='/workplace/certificate/
discussions.cer', type='TrustedCertificate')
```

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, as described in Registering Discussions Servers. Figure 28-7 shows example connection detail settings for the Edit Discussions and Announcement Connection page.

Figure 28-7 Edit Discussions and Announcement Connection Page



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: 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: 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: 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: Importing the Certificate

   ```
   keytool -importcert -alias producer_public_key -file producer_public_key.cer  -
   keystore webcenter.jks -storepass MyPassword
   ```

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 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.

This section includes the following topics:

- Configuring a Typical Topology for Applications Consuming WebCenter Portal Client API

- Configuring a Multiple Domain Topology for Applications Consuming the WebCenter Portal Client API

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("producer");
```

> **Note:**
>
> The alias here should always be the public key.

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 are exactly same as those in Creating the WebCenter Portal Domain Keystore, and you would then also need to specify the following on your client stub:

```
GroupSpaceWSContext context = new GroupSpaceWSContext();
context.setRecipientKeyAlias("producer");
```

Configuring a Multiple Domain Topology for Applications Consuming the 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("producer");
```

> **Note:**
>
> The alias here should always be the public key.

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 are exactly same as those in Creating the WebCenter Portal Domain Keystore, and you would then also need to specify the following on your client stub:

```
GroupSpaceWSContext context = new GroupSpaceWSContext();
context.setRecipientKeyAlias("producer");
```

JKS Command Summary for a Typical Topology

Use the following command summary to quickly configure the keystore for a typical topology. These commands explain how to configure a JKS keystore.

WebCenter Side

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
```

SOA Side

```
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
bpel.cer
keytool -importcert -alias webcenter_spaces_ws -file webcenter_public.cer -keystore
bpel.jks -storepass welcome1
```

WebCenter Side

```
keytool -importcert -alias bpel -file bpel.cer -keystore webcenter.jks -storepass
welcome1
```

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 SOA Domain Keystore

Run the following WLST command to register the keystore:

```
configureWSMKeystore('/WLS/WC_Domain',JKS, 'webcenter.jks',
signAlias='producer', signAliasPassword='signAliasPassword',
cryptAlias='cryptAlias', cryptAliasPassword='cryptAliasPassword')
```

Where:

- *WC_Domain* — TheWebCenter Portal domain
- *signAliasPassword* — The password for the public key
- *cryptAlias* — The public key alias
- *cryptAliasPassword* — The password for the public key

For example:

```
configureWSMKeystore(context='/WLS/WC_Domain', keystoreType='JKS',
location='./consumer.jks',keystorePassword='welcome1', signAlias='consumer',
signAliasPassword='welcome1', cryptAlias='consumer',
cryptAliasPassword='welcome1')
```

JKS Command Summary for Extensions to a Typical Topology

Use the following command summary to quickly configure the keystore and DF properties for a multi-domain topology.

WebCenter Side

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
```

SOA Side.

```
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 bpel.cer

keytool -importcert -alias webcenter_spaces_ws -file webcenter_public.cer -
keystore bpel.jks -storepass welcome1
```

When prompted to trust the certificate, say `yes`.

Discussions

```
keytool -genkeypair -keyalg RSA -dname "cn=disc,dc=example,dc=com" -alias
discussions -keypass MyPassword -keystore discussions.jks

keytool -exportcert -v -alias discussions -keystore discussions.jks -storepass
MyPassword -rfc -file disc.cer

keytool -importcert -alias webcenter_public -file webcenter_public.cer -keystore
discussions.jks -storepass MyPassword
```

When prompted to trust the certificate, say `yes`.

WebCenter Side

```
keytool -importcert -alias df_webcenter_public -file disc.cert -keystore
discussions.jks -storepass MyPassword
```

When prompted to trust the certificate, say `yes`.

```
keytool -importcert -alias webcenter_spaces_ws -file bpel.cer -keystore bpel.jks
-storepass MyPassword
```

> **Note:**
>
> Maintain the name of the alias as `'webcenter_spaces_ws'`.

Configure the External Discussions Server Domain Keystore

Run the following WLST command to register the keystore on the **WebCenter Side**:

```
configureWSMKeystore(context='/WLS/wc_domain',keystoreType='JKS',location='./
producer.jks',keystorePassword='welcome1',signAlias='producer',signAliasPassw
ord='welcome1',cryptAlias='producer',cryptAliasPassword='welcome1')
```

Where:

- *wc_domain* — TheWebCenter Portal domain
- *signAliasPassword* — The password for the public key
- *cryptAlias* — The public key alias
- *cryptAliasPassword* — The password for the public key

Configure the SOA Domain Keystore

Run the following WLST command to register the keystore:

```
configureWSMKeystore(context='/WLS/wc_domain',keystoreType='JKS',location='./
bpel.jks',keystorePassword='welcome1',signAlias='producer',signAliasPassword=
'welcome1',cryptAlias='producer',cryptAliasPassword='welcome1')
```

Where:

- *wc_domain* — TheWebCenter Portal domain
- *signAliasPassword* — The password for the public key
- *cryptAlias* — The public key alias
- *cryptAliasPassword* — The password for the public key

Registering Discussions keystore

Run the following WLST command to register the keystore:

```
configureWSMKeystore('/WLS/wc_domain','JKS', discussions.jks,
signAlias='producer', signAliasPassword='signAliasPassword',
cryptAlias='cryptAlias', cryptAliasPassword='cryptAliasPassword')
```

Where:

- *wc_domain* — TheWebCenter Portal domain
- *signAliasPassword* — The password for the public key
- *cryptAlias* — The public key alias
- *cryptAliasPassword* — The password for the public key

Configure the Discussions Server Connection

Supply the WS-Security client certificate information within the discussions server connection that is configured for WebCenter Portal, as described in Registering Discussions Servers. Also see Configuring the Discussions Server Connection Settings for example connection detail settings for the Edit Discussions and Announcement Connection page.

29

Configuring Security for Portlet Producers

Configure WebCenter Portal to handle security for WSRP and JPDK portlet producers.

> **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 Understanding Administrative Operations, Roles, and Tools.

Topics:

- Securing a WSRP Producer
- Securing a PDK-Java Producer

Securing a WSRP Producer

The following sections describe how to secure access to JSR-168 standards-based WSRP portlets from WebCenter Portal:

- Deploying the Producer
- Attaching a Policy to the Producer Endpoint
- Setting Up the Keystores

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 Deploying Portlet Producer Applications.

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 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.

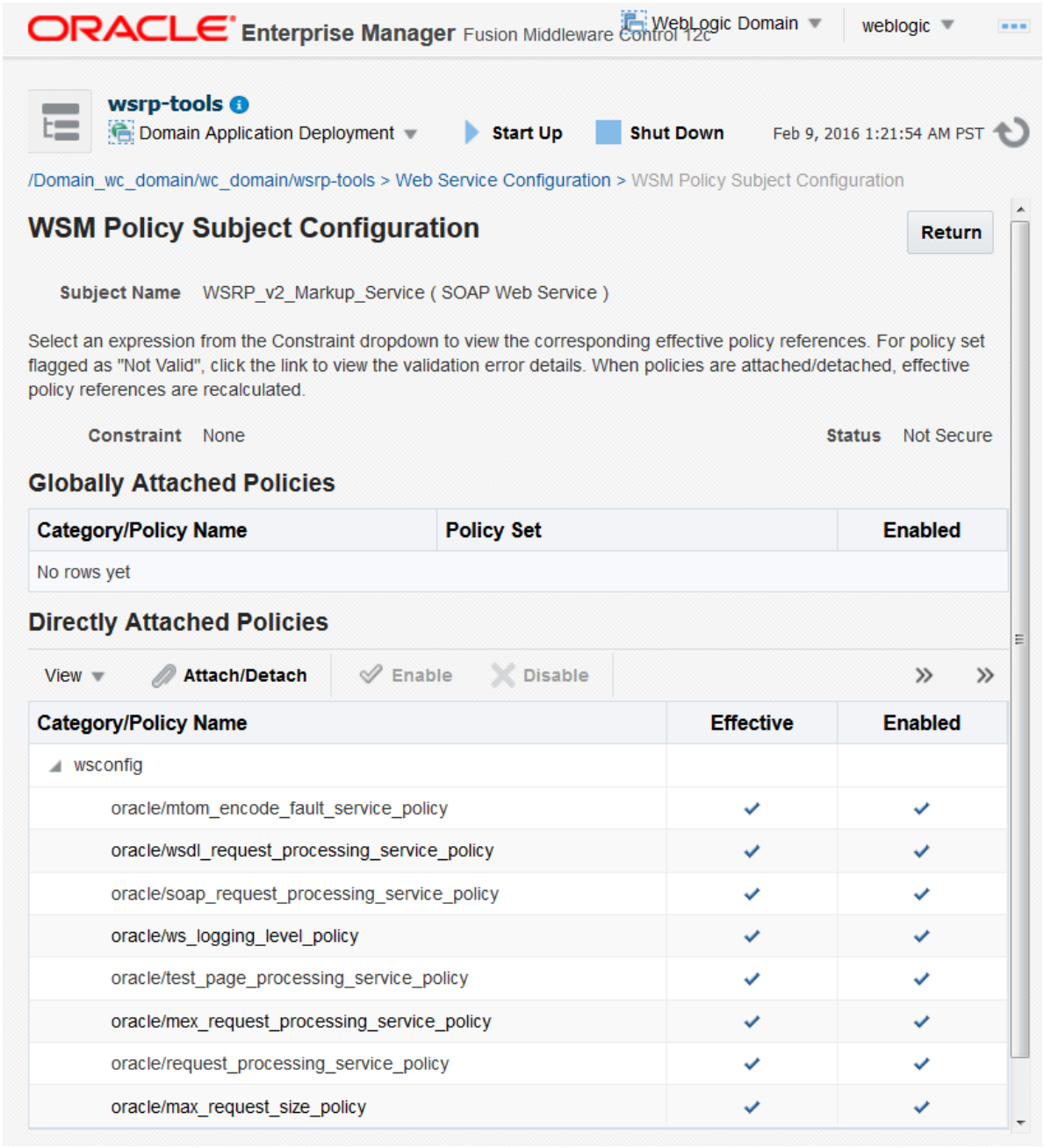
3. From the Application Deployment menu, select **Web Services**.

   The Web Services Summary page for the producer displays.

4. Open the Web Service Endpoint tab and click the endpoint to which to attach a policy.

   The WSM Policy Subject Configuration page displays (see Figure 29-1).

Figure 29-1 WSM Policy Subject Configuration



> **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.

5. Open the Policies tab to display the currently attached policies for the producer.

6. Click **Attach/Detach** to add or remove a policy.

   The Attach/Detach Policies page is shown listing the available policies and their descriptions.

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.

9. When finished adding polices to attach to the producer endpoint, click **OK**.

Setting Up the Keystores

The steps to create and configure keystores for a WSRP producer depend on the topology of your WebCenter Portal environment, and are covered in the following sections:

- Configuring WS-Security for a Typical Topology
- Configuring WS-Security for Multiple Domains

Refer to these sections for more complete instructions for setting up the keystores, and other WS-Security aspects of configuring WSRP producers.

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 Securing the WebCenter Portal Connection to Portlet Producers with SSL.

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:

- Defining a Shared Key Using Fusion Middleware ControlFusion Middleware Control
- Defining a Shared Key Using WLST

Defining a Shared Key Using Fusion Middleware ControlFusion Middleware Control

To define a shared key using Fusion Middleware Control:

1. Log into 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**, then **Credentials**.

   The Credentials pane displays.
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.

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 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, `provider1`)
   - *`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.provider1.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.

Registering an Oracle PDK-Java Producer with a Shared Key

Registering a PDK-Java producer is described in Registering an Oracle PDK-Java Portlet Producer. When you register an Oracle 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**.

30

Managing Impersonation

Manage and configure WebCenter Portal Impersonation, which lets designated WebCenter Portal users impersonate other 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 Using WebCenter Portal Impersonation in *Using Portals in Oracle WebCenter Portal*. For information about impersonation ELs and APIs, see ELs Related to Impersonation in *Developing for Oracle WebCenter Portal*.

> **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, Understanding Administrative Operations, Roles, and Tools.

Topics:

- Introduction to WebCenter Portal Impersonation
- Preparing WebCenter Portal for Impersonation
- Configuring WebCenter Portal for Impersonation
- Configuring Impersonators
- Disabling Impersonation
- Turning off the Session Indicator
- Overriding the Impersonation Hotkey
- Managing Audit Logs for WebCenter Portal Impersonation

Introduction to WebCenter Portal Impersonation

This section includes the following topics:

- About WebCenter Portal Impersonation
- Best Practices for Using WebCenter Portal Impersonation

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.

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 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.

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 11*g* (Oracle's single sign-on solution), and then turn on impersonation in OAM. For information about installing and configuring OAM 11*g*, see Configuring Oracle Access Manager.

This section includes the following topics:

- WebCenter Portal Impersonation Requirements
- Turning on Impersonation in OAM

- Adding Impersonation Attributes to the Identity Store

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 10*g* or 11*g* WebGate (see Configuring Oracle Access Manager)
- 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 Portal role

Turning on Impersonation in OAM

After installing and configuring OAM 11.1.2.0 (with either the 10*g* or 11*g* WebGate) as described in Configuring Oracle Access Manager, continue by enabling impersonation in OAM using `idmConfigTool` as shown below.

To enable impersonation:

1. Use `idmConfigTool` to configure OAM
2. Create the properties file as shown, but set `OAM11G_IMPERSONATION_FLAG` to `true`.

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 topics:

- Adding Impersonation Attributes for Individual Users
- Adding Impersonation Attributes for Multiple Users

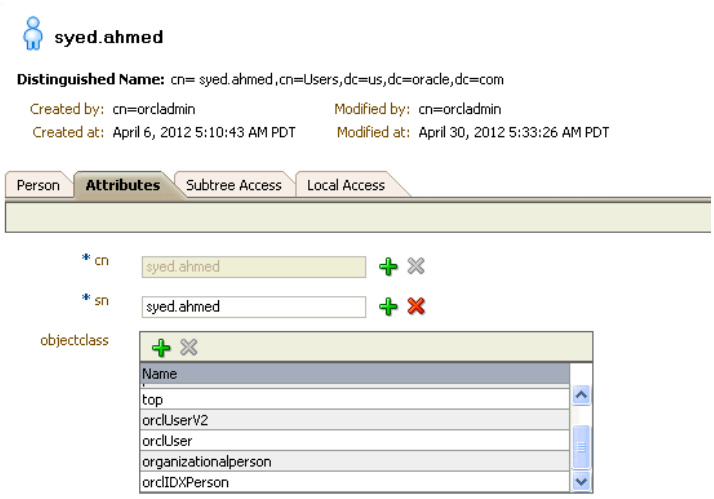
Adding Impersonation Attributes for Individual Users

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 30-1.

Figure 30-1 ODSM Attributes Tab



   c. Click **Apply**.

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 *Reference for Oracle Identity Management*.

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_Portal` 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_Portal`).

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 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.

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.

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 30-2.

> **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 30-2 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_Portal` 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 Using WebCenter Portal Impersonation in *Using Portals in Oracle WebCenter Portal*.
4. Restart the `WC_Portal` managed server for the change to take effect.

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. In the portal browser, click the **Administration** tile, then click **Attributes** in the left pane.

   > **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_Portal` server.

3. On the **Attributes** page, click **Add Attribute**, and set the new hotkey sequence as follows:

   `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_Portal` server for the change to take effect.

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 Managing WebCenter Portal Audit Logs. For information about configuring the Audit Service to use a database, see Configuring and Managing Auditing in *Securing Applications with Oracle Platform Security Services*.

Part VI

Administering WebCenter Portal Lifecycle

This part of *Administering Oracle WebCenter Portal* provides information about the WebCenter Portal lifecycle operations.

- Understanding the WebCenter Portal Lifecycle
- Deploying Portals, Templates, Assets, and Extensions
- Managing WebCenter Portal Backup, Recovery, and Cloning

31

Understanding the WebCenter Portal Lifecycle

To administer WebCenter Portal effectively, it is important to have a general understanding of the tasks, tools, and techniques for managing WebCenter Portal throughout its lifecycle.

> **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 WebCenter Portal Administration.
>
> See also, Understanding Administrative Operations, Roles, and Tools.

Topics:

- What Is the WebCenter Portal Life Cycle?
- What Are the Major WebCenter Portal Lifecycle Tasks?
- Permissions Required to Perform WebCenter Portal Lifecycle Operations
- Managing Security Through the WebCenter Portal Lifecycle

What Is the WebCenter Portal Life Cycle?

The portal life cycle describes the process of creating a portal using WebCenter Portal through deployment to a production instance. Many actors participate in the life cycle including software developers, content modelers, content contributors, IT administrators, and 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 31-1.

Table 31-1 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 WebCenter Portal's browser-based tooling 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 • Portal assets such as skins, page templates, and Content Presenter display templates • Visualization and custom visualization templates • shared libraries • data transfer and interportlet communication • initial security • Portlets
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. Components such as application data sources and portlet producers may be shared between the test and development environments.

Table 31-1 (Cont.) WebCenter Portal Life Cycle Phases

Life Cycle Phase	Primary Actors/Roles	Description
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`),and a collaboration server for discussions and announcements (`WC_Collaboration`). Also, an external LDAP-based identity store, such as Oracle Internet Directory, must be set up for the staging environment. The staging server is often maintained as a mirror of the production site. Occasional updates from development to portlets, task flows, and portal assets will need to be deployed to the stage environment. WebCenter Portal administration enables 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. **Note:** Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances
Production	• Application Specialists • System Administrators • Content Contributors • Knowledge Workers	A production portal is live and available to end users. Individual users with proper authorization can also customize their view. You can use WebCenter Portal administration to move portals and content to the production environment. Some back-end data must be moved manually. You can also use WLST commands for moving portals and content. Administrators can propagate portal changes in staging to production provided that the two environments are kept "in sync", that is, by always making changes in stage first and then pushing the changes to production using deployment or propagation. A portal in production can be modified whilst online in WebCenter Portal. However, changes made directly on the production server must be minimal.

What Are the Major WebCenter Portal Lifecycle Tasks?

Each phase of the lifecycle requires actors (developers, administrators, content contributors, and others) to perform certain tasks. This section provides an overview of the kinds of tasks that are performed during the portal lifecycle.

- One-Time Setup Tasks
- Understanding WebCenter Portal Staging and Production Environments
- Lifecycle Tasks

One-Time Setup Tasks

You must perform certain preparatory steps to set up development, test, stage, and production environments for WebCenter Portal. Table 31-2 provides a general list of these preliminary setup tasks and the environments to which they apply.

Table 31-2 Typical One-Time Setup Tasks

Setup Task	Development in JDeveloper (Assets and Shared Libraries only)	Development/ Test in WebCenter Portal	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

Understanding WebCenter Portal Staging and Production Environments

This section discusses the staging and production phases of the WebCenter Portal lifecycle. Figure 31-1 illustrates the general flow from staging to production environments. 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. This also involves migration of the policy store, MDS data (application integration, REST endpoints, SQL data sources), and all WebCenter Portal data stored on the Content Server repository.

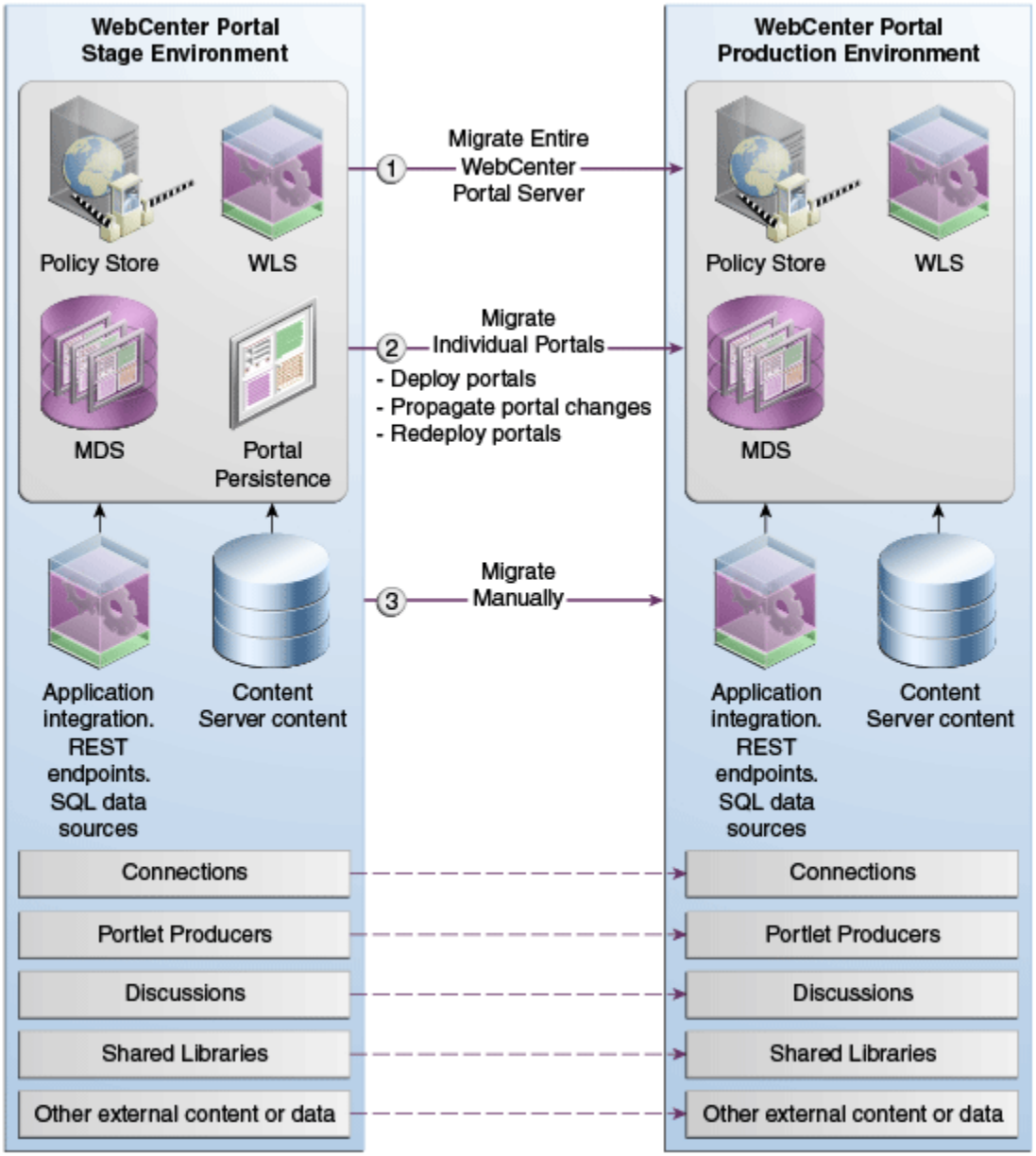
Subsequently, and once the production environment is live, you can propagate portal changes on production as and when required. Any new portals that are developed on stage can be individually deployed to production. Also, if required, you can redeploy existing portals.

You can manually move connections separately for all portals from one WebCenter Portal instance to another.

For information about the various lifecycle tasks that you perform on stage and production environments, see Lifecycle Tasks.

> **Note:**
>
> Figure 31-1 does not depict all possible portal features.

Figure 31-1 Flow from WebCenter Portal Staging to Production Environments



Lifecycle Tasks

Table 31-3 describes the tasks that you may need to perform in the WebCenter Portal lifecycle.

Table 31-3 Lifecycle Tasks

Lifecycle Task	Description	Tools Used	How To Do?
Migrate the entire portal instance	Once both staging and production environments are set up and configured, copy your WebCenter Portal instance on stage to the target for the first time.	• Enterprise Manager Fusion Middleware Control Console • WLST	• Migrating Entire WebCenter Portal to Another Target
Deploy portals	Deploy portals directly to the target or create a portal archive and import it on the target. You can also export individual production portals to an archive and import them back to your staging site.	• WebCenter Portal • WLST	• Deploying Portals
Deploy individual assets	You can share assets or migrate assets to other WebCenter Portal instances. You can also download assets and edit and extend them in tools such as Oracle JDeveloper, and then deploy them back to WebCenter Portal. Developers can deploy portal assets/extensions to WebCenter Portal directly from JDeveloper if they have the required permissions.	• WebCenter Portal • WLST	• Deploying Portal Templates • Deploying Assets
Propagate portal changes	You can propagate portal changes made in staging to production if your stage and production environments are connected and kept "in sync". In portal propagation, only the incremental changes made to a portal on the source are pushed to the target server.	• WebCenter Portal • WLST	• Propagating and Redeploying Portals in Production
Redeploy portals	After initial deployment of a portal, you can choose to redeploy the portal to the target. When you redeploy a portal, it is deleted and re-created as a new portal.	• WebCenter Portal • WLST	• Redeploying a Portal Using WebCenter Portal

Table 31-3 (Cont.) Lifecycle Tasks

Lifecycle Task	Description	Tools Used	How To Do?
Migrate connections	When you deploy a portal, its connection are also deployed. You can move connections for all portals separately from one WebCenter Portal instance to another.	• WLST	• Moving Connections Details from Staging to Production
Backup	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.	• WebCenter Portal • WLST	• Backing Up Individual Portals • Backing Up an Entire WebCenter Portal Installation
Recover	You can completely restore one or more portals or your entire WebCenter Portal installation from a backup archive.	• WebCenter Portal • WLST	• Restoring Portals from a Backup • Restoring an Entire WebCenter Portal Installation • Restoring WebCenter Portal from Backups Using Scripts

Permissions Required to Perform WebCenter Portal Lifecycle Operations

Table 31-4 describes which WebLogic Server roles and WebCenter Portal permissions are required to perform lifecycle operations.

Table 31-4 WebCenter Portal and WebLogic Server Permission Requirements for Lifecycle Operations

WebCenter Portal Object	Tool	WebLogic Server Role	WebCenter Portal Permission
WebCenter Portal (application level)			
Import or export archive	Fusion Middleware Control	`Monitor` (or higher)	`Application: Manage`
Import or export archive	WLST	`Monitor` (or higher)	`Application: Manage`
Portal — Direct deployment/Archive import			

Table 31-4 (Cont.) WebCenter Portal and WebLogic Server Permission Requirements for Lifecycle Operations

WebCenter Portal Object	Tool	WebLogic Server Role	WebCenter Portal Permission
Directly deploy a portal or import a portal archive	WebCenter Portal		`Portal Server - Deploy` (source) `Portals: Manage Configuration` (source) `Portals - Create Portals` (target)
Directly deploy a portal or import a portal archive	WLST	`Monitor` (or higher)	`Portal Server - Deploy` (source) `Portals: Manage Configuration` (source) `Portals - Create Portals` (target)
Redeploy or propagate a portal or re-import a portal archive	WebCenter Portal		`Portal Server - Deploy` (source) `Portals: Manage Configuration` (source) `Portals: Manage Security and Configuration` (target) `Portals - Create Portals` (target)
Redeploy or propagate a portal or re-import a portal archive	WLST	`Monitor` (or higher)	`Portal Server - Deploy` (source) `Portals: Manage Configuration` (source) `Portals: Manage Security and Configuration` (target) `Portals - Create Portals` (target)
Portal — Export an Archive			
Export a portal archive	WebCenter Portal	-	`Portals: Manage Security and Configuration`
Export a portal archive	WLST	`Monitor` (or higher)	`Portals: Manage Security and Configuration`
Portal Template			
Export or import a portal template archive	WebCenter Portal	-	`Portal Templates: Manage All`
Export or import a portal template archive	WLST	`Monitor` (or higher)	`Portal Templates: Manage All`
Portal Asset			
Export or import an asset archive	WebCenter Portal/REST API	-	`Portal: Manage Configuration` And either: • `Create, Edit, Delete Assets` • `Create, Edit, Delete <Portal_Asset_Type>`
Export or import an asset archive	WLST	`Monitor` (or higher)	Either: • `Create, Edit, Delete Assets` • `Create, Edit, Delete <Portal_Asset_Type>`
Shared Library			

Table 31-4 (Cont.) WebCenter Portal and WebLogic Server Permission Requirements for Lifecycle Operations

WebCenter Portal Object	Tool	WebLogic Server Role	WebCenter Portal Permission
Deploy portal extension directly from JDeveloper	JDeveloper	`Monitor` (or higher)	`Portals: Manage All`
WebCenter Portal Connections			
Export or import all WebCenter Portal connections	WLST	`Operator` (or higher)	-
Shared Asset			
Import or export asset archive	WebCenter Portal	-	`Application: Manage Configuration` `Create, Edit, Delete <Shared_Asset_Type>`
Import or export asset archive	WLST	`Monitor` (or higher)	`Create, Edit, Delete <Shared_Asset_Type>`

Managing Security Through the WebCenter Portal Lifecycle

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, 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 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 Backing Up and Restoring Policy Stores (LDAP and Database) and Backing Up and Restoring Credential Stores (LDAP and Database).

32

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 instances and stage or production environments.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

> **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 Permissions Required to Perform WebCenter Portal Lifecycle Operations.
>
> See also Understanding Administrative Operations, Roles, and Tools.

Topics:

- Deploying Portals
- Deploying Portal Templates
- Deploying Assets
- Deploying Custom Shared Library Extensions
- Moving Connections Details from Staging to Production
- Migrating Discussions Resources for a Portal
- Propagating and Redeploying Portals in Production

Deploying Portals

This section includes the following topics:

- About Portal Deployment

- Directly Deploying Portals Using WebCenter Portal
- Directly Deploying Portals Using WLST
- Deploying Portal Archives

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 choose to include *all or some* of the source portal's data.

After initial deployment of a portal, you can choose to redeploy the portal or propagate only portal changes to the target. When you redeploy a portal, it is simply deleted and re-created as a new portal. In portal propagation, only the incremental changes made to a portal on the source are pushed to the target server.

You can deploy a portal in the following ways:

- **Direct portal deployment** - If a direct connection to the target server exists, you can deploy a portal to the target server by using WebCenter Portal Administration. You can also use the `deployWebCenterPortal` WLST command to deploy portals directly to the target server. For details, see Directly Deploying Portals Using WebCenter Portal and Directly Deploying Portals Using WLST.
- **Portal archive deployment** - You can export the archive (`.par` file) of the source portal and import the archived portal on the target server by using WebCenter Portal Administration. You can also use WLST commands to export portals to an archive and then import portals from the file.

  For details, see Exporting and Importing Portal Archives.

Information Always Deployed with a Portal

When a portal is deployed, the following details are always included:

- Portal pages
- Portal assets: Page templates, resource catalogs, skins, page styles, Content Presenter display templates, task flow styles, task flows, layouts, data controls, visualization templates, data visualizations (including dependant business objects and data sources), business objects, data sources (including their connections)
- Portal activity/usage data: Activity streams, calendar events, feedback, lists, links, message boards, people connections, profiles, and surveys
- Portal security data: Portal roles and permissions and member details and their role assignments

Information that can be Optionally Deployed with a Portal

When deploying a portal, you can optionally choose to include the following as part of portal deployment:

- Portal's content: A portal's documents and associated content are placed in the portal's content folder on Content Server. If you choose not to move the content folder during portal deployment, you can manually move the folder to the target using WebCenter Content Server migration tools. For details, see System Migration and Archiving in *Administering Oracle WebCenter Content*.

  Portal deployments do not include the content that is stored outside a 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's content folder, you must either manually move such content to the target or ensure that the target can access the same content as the source. When you move a portal to a different target, Content Presenter data references are maintained only if Content Server connection names and root folder names are the same in both the source and the target.

- Shared assets: While deploying your portal you can choose to deploy all shared assets used by the portal.
- Shared library: While deploying your portal you can choose to deploy the shared libraries used by the portal. When you choose to deploy shared libraries, the main shared library that gets deployed is `extend.spaces.webapp`, which in turn may be dependent on other libraries. As part of deployment, all new versions (newly created or updated) of the dependent libraries of the main shared library are also included. However, this is done only for the first level of dependent libraries. For example, suppose `extend.spaces.webapp` is dependent on `CustomSharedLibrary1`, and `CustomSharedLibrary1` is dependent on `CustomSharedLibrary2`. If an updated version is available for both `CustomSharedLibrary1` and `CustomSharedLibrary2`, only `CustomSharedLibrary1` is included as part of shared library deployment.

Information Not Included During Portal Deployment

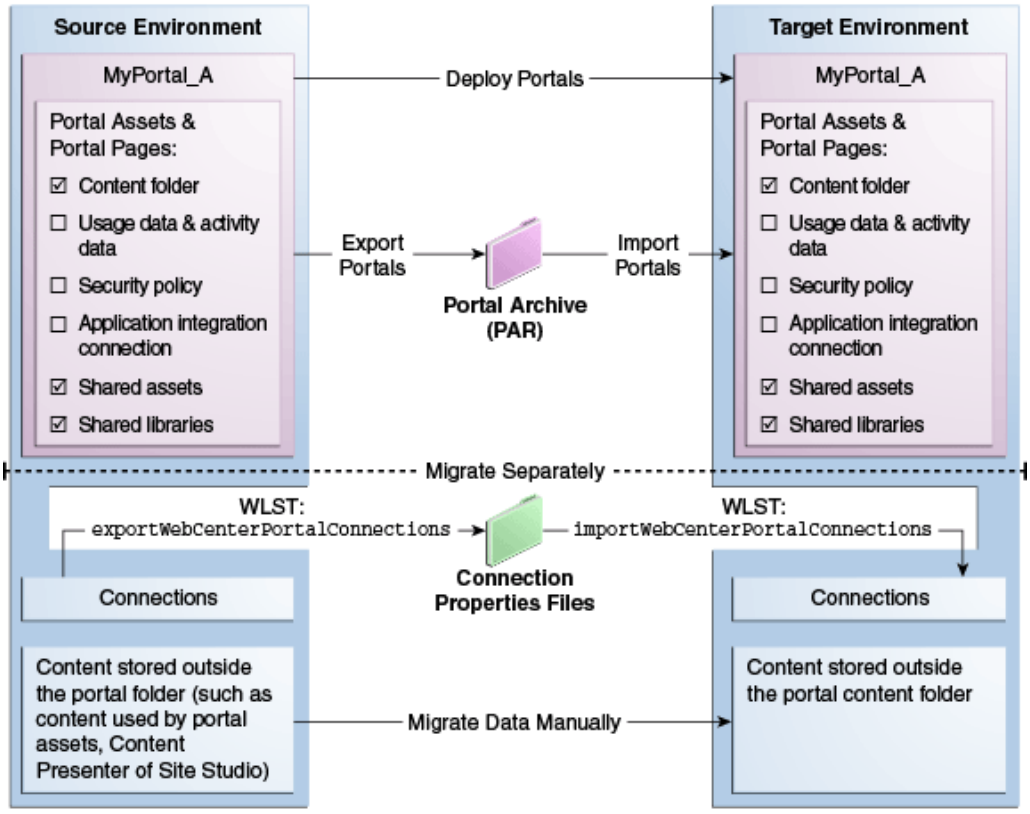
Some portal information is stored externally and cannot be deployed at the same time as the portal, for example:

- content used by portal assets, Content Presenter or Site Studio stored outside of the portal's content folder
- portal discussions
- portal mail
- portal analytics

> **Note:**
>
> Connections are exported and imported separately. For more information, see Understanding Connection Property Files.

Figure 32-1 illustrates the different ways in which you can move a portal (and its associated data) to another server.

Figure 32-1 Deploying Portals to a Target Server



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.

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.

> **Note:**
>
> While exporting or deploying a portal if the server goes down and fails over to another server in the cluster, the operation will fail. You need to refresh the page and perform the export or deployment operation again.
> If you want to deploy a portal larger than 50 MB, ensure that you modify the maximum file upload size on the target server as per your requirements. For information, see Modifying the File Upload Size in Content Manager.

For information about troubleshooting portal deployment issues, see Troubleshooting WebCenter Portal.

Directly Deploying Portals Using WebCenter Portal

Using WebCenter Portal administration, you first create a connection to the target server and then directly deploy your portals to the target server. After a portal is deployed, you can view its deployment status and deployment history.

This section includes the following topics:

- Creating a Portal Server Connection
- Deploying a Portal Using WebCenter Portal
- Viewing Portal Deployment History

Creating a Portal Server Connection

Before you can deploy a portal, you need to set up a connection to the target portal server.

To create a portal server connection:

1. Log on to WebCenter Portal, and navigate to portal administration.
2. Click **Tools and Services**.
3. Select **Portal Server Connections** from the list of tools and services.
4. Click **Create**.
5. In the Create Portal Server Connection page, specify the following details:
   a. **Name**: Specify the name of the connection. Note that only alphanumeric characters can be used.
   b. **URL**: Specify the URL of the target portal server in the following format:

      `http://targetserverhost:port`

      where *`targetserverhost:port`* refer to the host name and port number of the portal server where you want to deploy your portals.
   c. **Username**: Type the user name used for connecting to the target server.
   d. **Password**: Type the password for the specified user name.
6. Click **Test** to make sure the connection works.
7. Click **Create**.

   Note that if the connection test fails due to the portal server being offline, the connection will still be set up, and can be used once the server is available.

Deploying a Portal Using WebCenter Portal

> **Note:**
>
> Deploying a portal is primarily a system administrator task; however, you can assign the `Portal Server: Deploy` permission to another custom role. It is recommended that you create a custom role and assign this permission to the custom role in order to restrict the user roles that can deploy portals.
>
> Only the Portal Manager (or Delegated Manager) of the portal can deploy the portal, and in addition, must be in a role that has the `Portal Server: Deploy` permission.

For more information about creating a custom application role and adding users to the role, see Defining Application Roles. and Assigning Users (and Groups) to Application Roles.

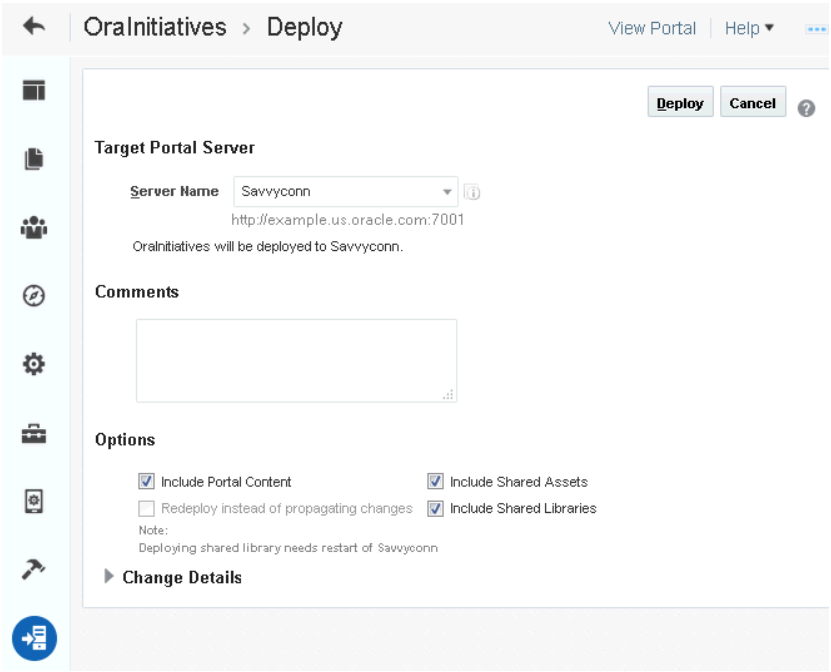
To deploy a portal using WebCenter Portal:

1. In WebCenter Portal, access portal administration as described in Accessing Portal Administration in *Building Portals with Oracle WebCenter Portal*
2. Click the **Deploy** icon.
3. From the **Server Name** list under **Target Portal Server**, select the portal server connection you want to use to deploy your portal.

   You created this connection as described in Creating a Portal Server Connection.
4. In the **Comments** box, specify comments, if any, about portal deployment.
5. In the Options section, select the deployment options:
   - **Include Portal Content**: Select to specify that the portal content stored on Content Server must be included in portal deployment on the target server.
   - **Include Shared Assets**: Deploys the shared assets used by the portal. Clear the check box if you do not want to deploy shared assets.
   - **Include Shared Libraries**: Deploys the shared libraries used by the portal. Clear the check box if you do not want to deploy shared libraries. If you include shared libraries in portal deployment, you must restart the target server after deploying the portal for the shared library changes to be picked up.

   If this is the first time the portal is being deployed, the **Redeploy instead of propagating changes** check box appears disabled. Expanding the **Change Details** section displays a message that the portal is being deployed (for the first time) and hence all the data will be carried over to the target server. When you propagate a portal, this section displays the changes made to the portal since the last deployment.

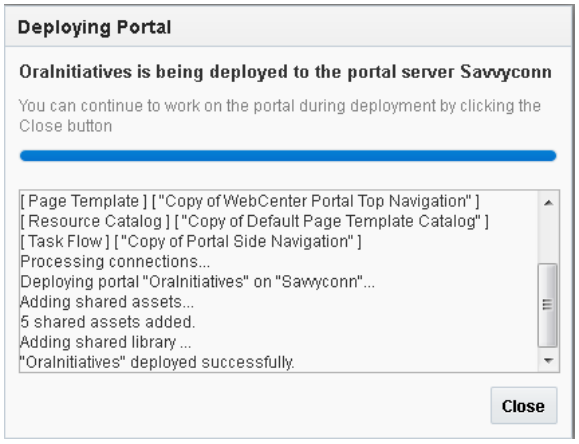
Figure 32-2 Deploying a Portal



6. Click **Deploy**.

   The Deploy Portals dialog displays the progress and status of portal deployment. While the portal is being deployed, you can choose to close the dialog and continue to work on the portal if required.

Figure 32-3 Portal Deployment Status



7. Click **Close**.

   Once a portal is deployed, you can view its deployment history and status.

   See Viewing Portal Deployment History.

8. Restart the target server where the portal is deployed if you included shared libraries in portal deployment.

   See Starting and Stopping Managed Servers for WebCenter Portal Application Deployments.

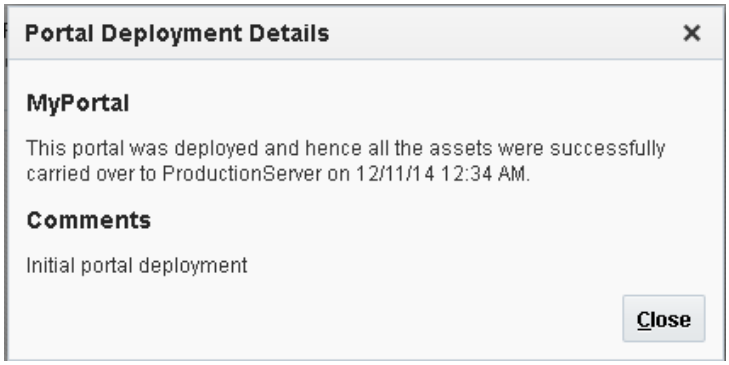
Viewing Portal Deployment History

To view portal deployment history using the WebCenter Portal Administration:

1. On the **Portals** administration page, click **Portal Deployments**.
2. On the **Recent Deployments** tab, click the **Details** link next to a portal to view deployment details.

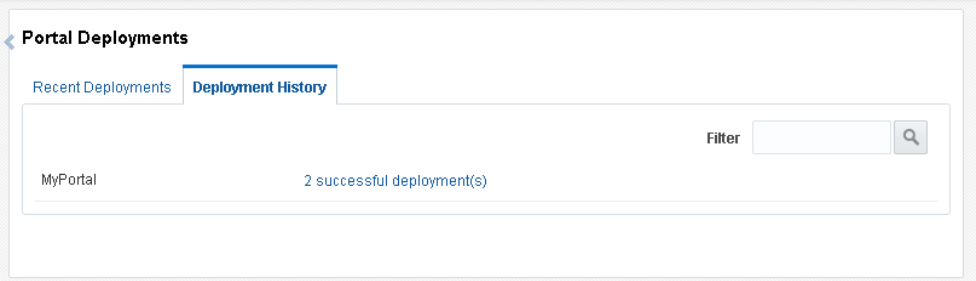
   The Portal Deployment Details dialog displays the name of the target server and the date and time of deployment. It also shows the comments, if any, added while deploying the portal.

Figure 32-4 Portal Deployment Details



3. Click **Close**.
4. To view each portal's deployment status, click the **Deployment History** tab.

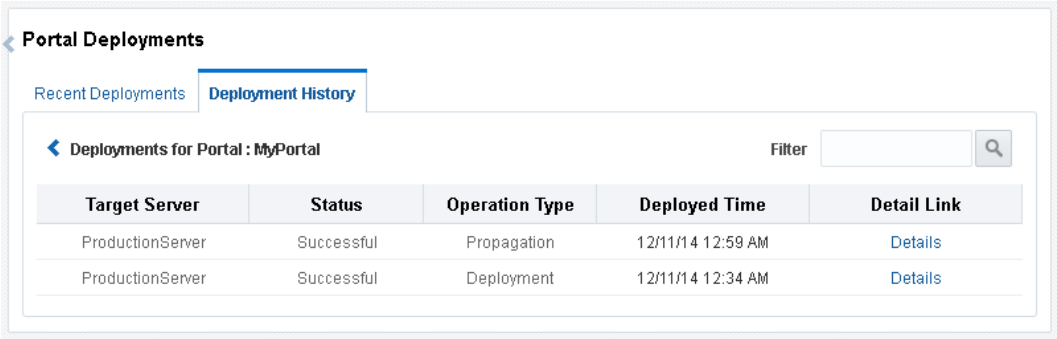
Figure 32-5 Deployment History



5. Click the deployment status link for a portal to display its deployment operations.

   Figure 32-6 shows the two records for the portal named `MyPortal`, one for initial deployment and the other for propagation.

Figure 32-6 Deployment Operations



6. Click **Details** next to a deployment operation to display more details.

Directly Deploying Portals Using WLST

You can use the WLST command `deployWebCenterPortal` to deploy a single, online portal directly to another target server. If you want to propagate portal changes in the source to the target using WLST, then you *must* use `deployWebCenterPortal` to deploy your portal.

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_Portal`.

   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 Exporting Portals to an Archive.

2. Verify that you have at least the WebLogic Server `Monitor` role and the WebCenter Portal permission `Portals: Manage Security and Configuration`.

3. Ensure a connection exists between the source and target WebCenter Portal. If a connection created using WebCenter Portal Administration already exists, you can use it to deploy portals or you can use the UI to create a new one.

   If a connection to the target from the source environment does not exist and you want to create one using WLST, use the WLST command `adf_createHttpURLConnection`.

   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 Running Oracle WebLogic Scripting Tool (WLST) Commands.

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 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, deployPortalContent, deploySecurity, deployData,
 deployActivities, deploySharedAssets, deployConnections, overwrite,
savePortal, deployLog, server,
 applicationVersion])
```

   For detailed command syntax and descriptions, see deployWebCenterPortal in *WebCenter WLST Command Reference Reference*. The options that you set depend on your specific deployment requirements.

   The following example deploys a new portal named `myPortal` for the first time on the target server. It also deploys all its associated content and specifies a name and location for the deploy log file:

```
deployWebCenterPortal(appName='webcenter',portalName='myPortal',
targetConnectionName='MyWebCenterPortalTarget',
deployPortalContent=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.

Redeploying a portal that exists on the target

If you want to redeploy a portal that already exists on the targer server, you use the `deployWebCenterPortal` command, with `overwrite=1`. The following example backs up a portal named `myExistingPortal` on the target and then overwrites the target portal (`overwrite=1`) with the source portal. The content associated with the target portal is preserved:

```
deployWebCenterPortal(appName='webcenter',portalName='myExistingPortal',
targetConnectionName='MyWebCenterPortalTarget',
deployPortalContent=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

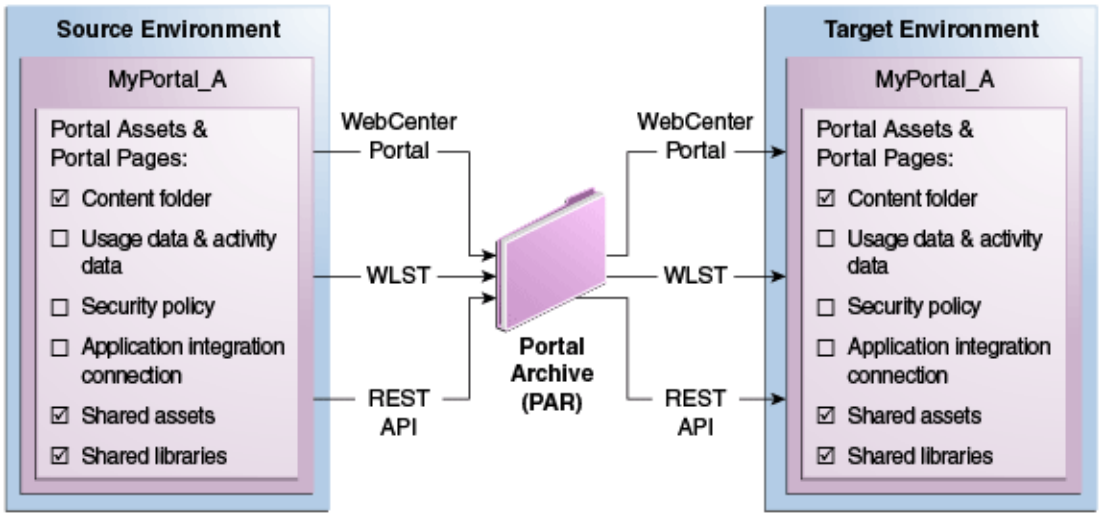
In the target WebCenter Portal:

1. Log in to the target WebCenter Portal.
2. Navigate to the new portal deployment.
3. Verify that the portal works as expected.

Deploying Portal Archives

Administrators can use WebCenter Portal or WLST commands to deploy portal archives (`.par` files) to any WebCenter Portal installation. The target portal server must be up and running when you deploy (or import) one or more portals from a file.

Figure 32-7 Deploying Portal Archives



This section includes the following topics:

- Understanding Portal Archives
- Securing Archives
- Exporting and Importing Portal Archives
- Exporting Portals to an Archive
- Importing Portals from an Archive
- 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 Propagating and Redeploying Portals in Production.

Understanding Portal Archives

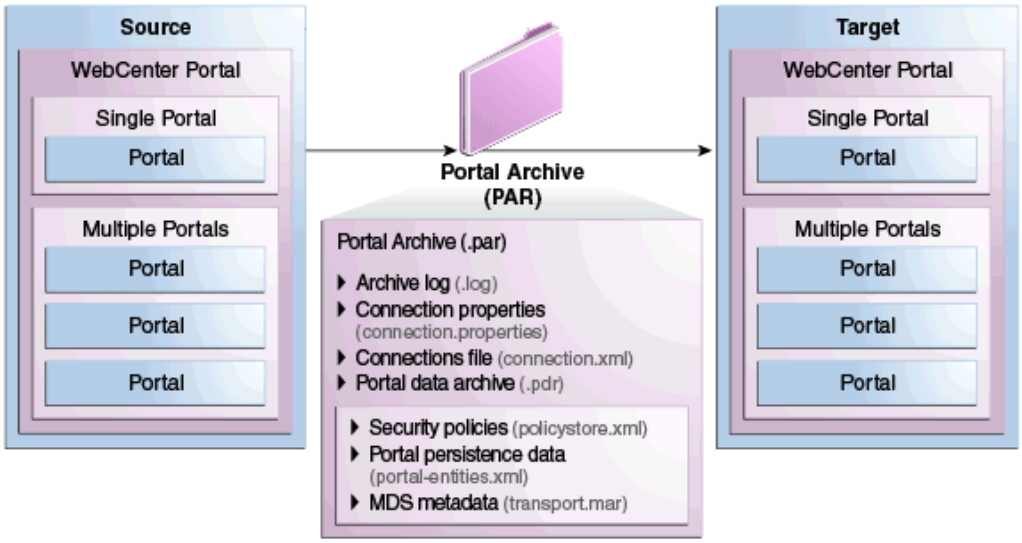
You can create a portal archive (`.par` file) for a single portal or you can archive multiple portals in the same `.par` file.

Portal archives can contain:

- One or more portal data archive (`.pdr`) files: Portal archives include a PDR for each portal that you add to the archive. A PDR includes the security policy for the portal and a metadata archive that captures metadata, data, and content for the portal.
- An export log file (`.log`): An export log file lists all the portals, MDS metadata files, and data (names of database tables that contain portal data) included in the archive.
- The connections.xml file For more information, see Understanding Connection Property Files.
- A WebCenter Portal connection properties file (`connection.properties`)

> **Note:**
>
> You can extract any portal archive (`.par` file) using the `listWebCenterPortalArchive` WLST command.

Figure 32-8 Portal Archive Deployment



Understanding Connection Property Files

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 Importing New WebCenter Portal Connections from a File.

> **Note:**
>
> - A `connection.properties` file is also generated when you run the WLST command `exportWebCenterPortalConnections`. For details, see, 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.

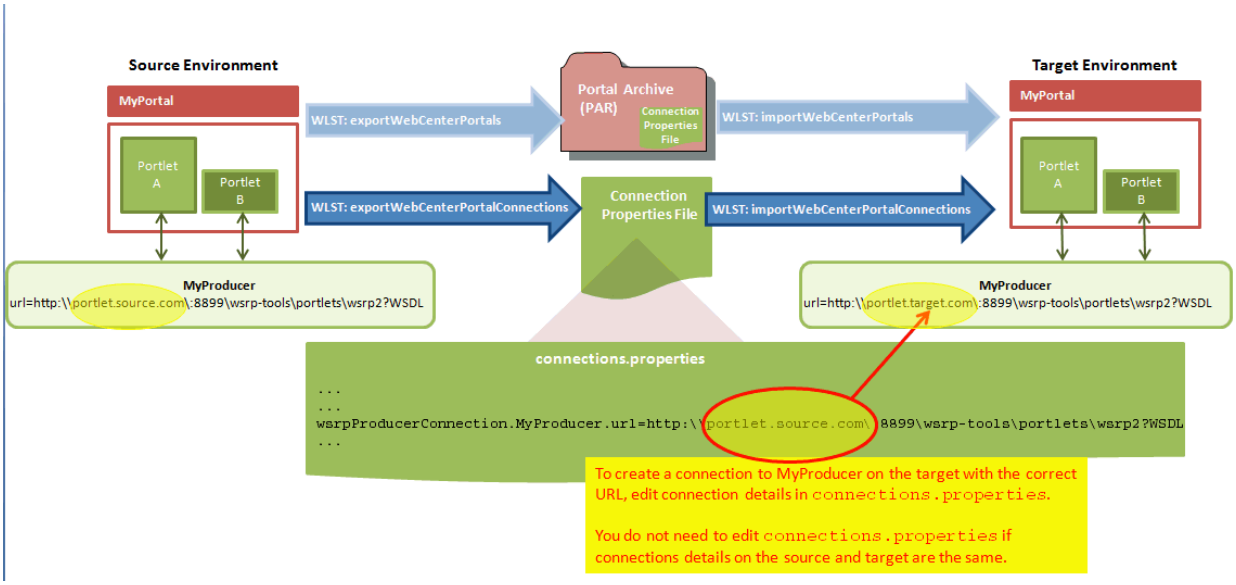
Table 32-1 shows examples where a different URL parameter is required on the target because the source and target do not use the same host.

Table 32-1 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 32-9 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 32-9 Using connection.properties to Create Connections on the Target



Connection Types and Connection Properties

Table 32-2 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 Administering Tools and Services .
> - Oracle strongly recommends that you only edit properties in `connection.properties` that are marked **Edit on Deployment?=Yes** in Table 32-2. 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 32-2 Connection Properties Exported to connection.properties

Connection Type	Properties Exported	Edit on Deployment?	Notes and Post Deployment Configuration Requirements
WSRP portlet producer	url proxyHost proxyPort timeout externalApp tokenType defaultUser issuerName recipientAlias	Yes Yes Yes No No No No No No	Security configuration post deployment: registrationProperties keyStorePath keyStorePswd sigKeyAlias sigKeyPswd encKeyAlias encKeyPswd enforcePolicyURI 1
PDK-Java producer	url proxyHost proxyPort subscriberId serviceId sharedKey timeout establishSession externalApp	Yes Yes Yes No No No No No No	Security configuration post deployment: mapUser useProxy
Web service connection	url proxyHost proxyPort mtom addressing wsrm security	Yes Yes Yes No No No No	Web service connections are used by data controls 1
URL connections - HTTP URL	url authenticationType connectionClassName realm	Yes No No No	Security configuration post deployment: password user attributes
URL connections - File URL	url	Yes	

Table 32-2 (Cont.) Connection Properties Exported to connection.properties

Connection Type	Properties Exported	Edit on Deployment?	Notes and Post Deployment Configuration Requirements
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	
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 Lync 2010	url	Yes	
	poolName	Yes	
	userDomain	Yes	
	adapter	No	
	timeout	No	
	appId	No	
	AdditionalProperty	No	
	defaultConnection	No	

Table 32-2 (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	
Search	url	Yes	Users will be prompted for appPassword if promptForPassword is set to 1
	appUser	No	
	defaultConnection	No	
	indexAliasName		
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 32-2 (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	
File System	path	Yes	
Worklist connection	BPELConnection	No	
Rest Connection	url	Yes	

[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 Moving Connections Details from Staging to Production.

Securing Archives

This section includes the following topics:

- About Securing Archive Files
- Securing Archive Files

About Securing Archive Files

WebCenter Portal supports validation checks to be performed when portal archives are exported or imported. This secures portal archives by preventing corrupt or arbitrary files from being included on Portal Server.

You can choose to set any of the following security levels for lifecycle operations:

- High-security mode: Set the `ExternallySecureLifecycleOperations` custom attribute, mapped against an external application that stores the credentials to encrypt or decrypt the file storing the checksum value.
- Moderate-security mode: Set the `SecureLifecycleOperations` custom attribute, mapped against the value `enable`. If the value is set to `enable`, lifecycle operations are secured, and checksum acts as the credentials for the encrypt or decrypt process.

During the export operations, a checksum is calculated and added to all the lifecycle archives - portal archives, application archives, and asset archives. The checksum is stored in a file named `lifecycle.chk` inside the `.par` or `.aar` files and the file is encrypted. During archive import, the file is decrypted to fetch the checksum value. In moderate security mode, the checksum (calculated internally) acts as the password to encrypt or decrypt the file. In high-security mode, the external application with shared credentials is used as the password, and the `ExternallySecureLifecycleOperations` custom attribute is used to fetch the password.

Consider the following while configuring a secure mode for lifecycle operations:

- If an archive is exported in a secured source environment, and the target environment is not secured, security validations are not performed on the archive.
- If lifecycle operations are secured, and during archive import if the`lifecycle.chk` file is missing from the archive, it is a security violation and the import operation is not allowed.
- If lifecycle operations need to be secured, the same level of security must be set on the target and the source instances. Different levels of security modes are not supported.
- If the high-security mode is set, both the source and the target instances must use the same password for the encryption and decryption to work.

Securing Archive Files

To secure application, portal, and asset archives, you can choose to set either the high-security mode or the moderate-security mode for lifecycle operations.

To secure your application, portal, and asset archives:

1. Log on to WebCenter Portal.
2. Configure the desired security mode to secure your archives:

Option	Procedure
To enable the high-security mode	a. Create a custom global attribute named `ExternallySecureLifecycleOperations`. b. Set the value of the custom attribute to `enable`. For information about creating a custom global attribute, see Adding a Global Attribute.
To enable the moderate-security mode	a. Register an external application and specify shared credentials. For information, see Registering External Applications. b. Create a custom global attribute named `SecureLifecycleOperations`. Specify the name of the external application as the value. For information about creating a custom attribute, see Adding a Global Attribute.

Exporting and Importing Portal Archives

If a direct connection to the target server does not exist, you can first export a portal to an archive (`.par` file) and then import the archive on the target server to deploy the portal. You can also create a portal archive if you want to create a backup of the portal and restore it on the same instance later.

> **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 Propagating and Redeploying Portals in Production.

To export and then import a portal archive:

1. Complete the portal archive prerequisites described in Portal Export Prerequisites.
2. Export the source portal:
   - To use WebCenter Portal, see Exporting Online Portals to an Archive Using WebCenter Portal Administration.
   - To use WLST, see Exporting Online Portals to an Archive Using WLST.
   - To use REST API, see Exporting a Portal Using REST APIs
3. (Optional) Migrate externally stored data and content to the target:

   For details, see Migrating Discussions Resources for a Portal.
4. Import the portal on the target:
   - To use WebCenter Portal, see Importing a Portal from an Archive Using WebCenter Portal Administration.
   - To use WLST, see Importing a Portal from an Archive Using WLST.
   - To use REST API, see Importing a Portal Using REST APIs

Exporting Portals to an Archive

You can generate an archive (`.par` file) for any portal that is running on WebCenter Portal. You can create a portal archive by using WebCenter Portal, the `exportWebCenterPortals` WLST command, or REST APIs.

To find out how to create portal archives, see:

- Portal Export Prerequisites
- Exporting Online Portals to an Archive Using WebCenter Portal Administration
- Exporting Online Portals to an Archive Using WLST
- Exporting a Portal Using REST APIs

Portal Export Prerequisites

Before exporting a portal to an archive (`.par` file), verify the following:

- **Portal content stored on Content Server** - If you want to include the portal content stored on Content Server in the portal archive, ensure Content Server is up and running. The portal archive does not include any security settings. On import, the security group of the target portal server is applied.
- **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_Portal',
>  toLocation='/tmp/content',
>  docs='/oracle/webcenter/siteresources/scopedMD/shared/**')
>
> importMetadata(application='webcenter', server='WC_Portal',
>  fromLocation='/tmp/content',
>  docs='/oracle/webcenter/siteresources/scopedMD/shared/**')
> ```

Exporting Online Portals to an Archive Using WebCenter Portal Administration

With `Portal Server-Manage All` or `Manage Configuration` permission, you can export portals to an archive using WebCenter Portal Administration, saving the portal archive to a 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.
>
> See Exporting Portal Templates to an Archive Using WebCenter Portal.

To export one or more portals:

1. On the **Portals** administration page (see Accessing the Portals Page in WebCenter Portal Administration), select the portal you want to export by highlighting the row in the table.

   Press Ctrl+click to select more than row.

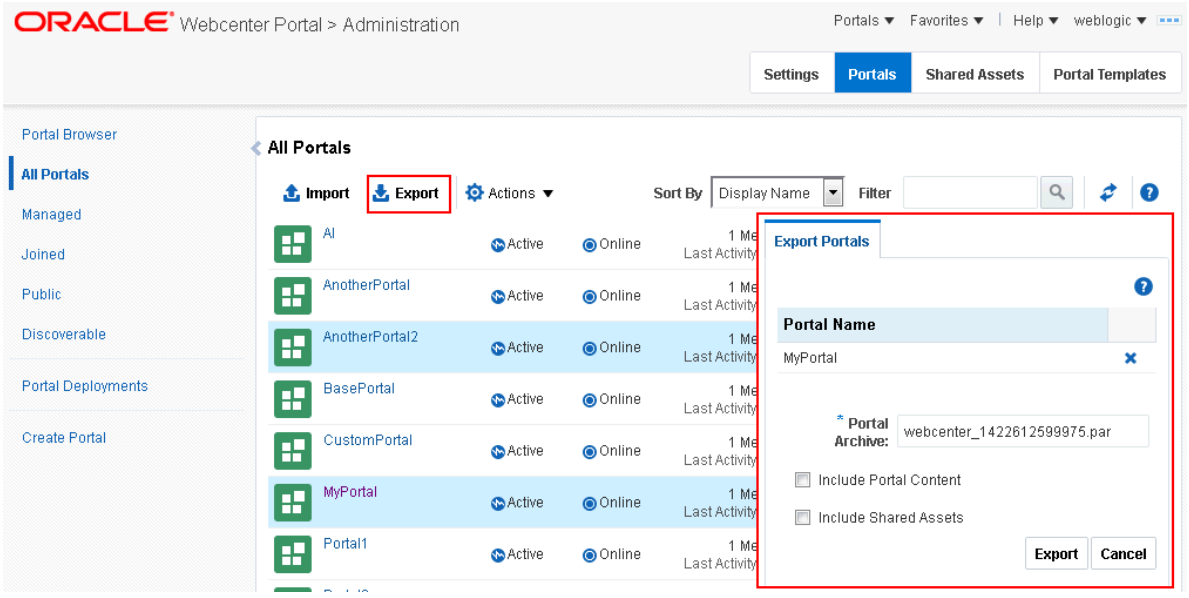
> **Note:**
>
> To prevent data conflict during the export process, Oracle recommends that all the portals you select are *offline* during the export process, even if only temporarily.
>
> See Taking Any Portal Offline.
>
> Members with the `Portals: Manage Security and Configuration` permission can still access a portal when it is offline, so notify them to not make changes while you complete the export.

2. Click **Export** in the toolbar.

   The Export Portals pane opens. All the portals that you select are listed.

   If you want to exclude a portal, click the **Delete** icon next to the portal's name.

Figure 32-10 Exporting Portals



3. 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`

4. Select **Include Portal Content** 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.

Note:

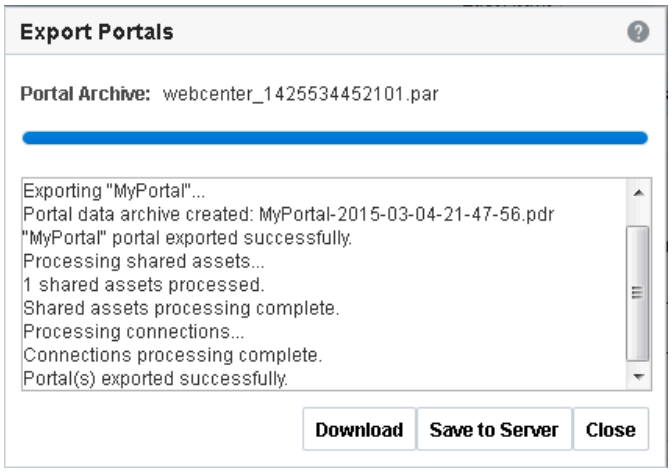
- The portal archive does not include any security settings. On import, the security group of the target portal server is applied.
- Including content folders increases the size of the portal archive. If you are exporting a large number of portals or large content folders, make sure that your portal archive size does not exceed the maximum upload limit of 2 GB.
- 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_Portal',
 toLocation='/tmp/content',
 docs='/oracle/webcenter/siteresources/scopedMD/shared/**')

importMetadata(application='webcenter', server='WC_Portal',
 fromLocation='/tmp/content',
 docs='/oracle/webcenter/siteresources/scopedMD/shared/**')
```

5. Select **Include Shared Assets** to export the shared assets used in the portal.
6. Click **Export**.

   Progress information displays during the export process.

Figure 32-11 Portal Export In Progress



7. Specify a location for the export archive (`.par` file) when the export process is complete.

   Select either of the following:

- **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. The .par file is saved to the default path *`DOMAIN_HOME`*`/WC_Archives`, where *`DOMAIN_HOME`* refers to the domain location where WebCenter Portal is installed.

  When the file is saved, click **OK** to close the Information dialog.

8. Click **Close**.

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, exportSharedAssets, server,
applicationVersion])
```

The options that you set depends on your specific archive requirements. For command syntax, see`exportWebCenterPortals` in *WebCenter WLST Command Reference Reference*.

For information on how to run WLST commands, see 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)
```

Exporting a Portal Using REST APIs

You can generate a portal archive (`.par` file) for your portals using the REST API support.

To export a portal using REST API, use the following URL format:

```
http://host:port/rest/api/v1/portal/portals/portal_shortId/archive?
utoken=utoken_value
```

Where *host*: *port* are the hostname and port number for the server where the portal is running, and *portal_shortId* is the short ID of the portal to be exported.

If you want to export the portal content as well, include the `includePortalContentValue` parameter in the URL as follows:

```
http://host:port/rest/api/v1/portal/portals/portal_shortId/archive?
includePortalContent=includePortalContentValue&utoken=utoken_value
```

The default value for `includePortalContent` is 0. Any value greater than 0 will be treated as true, and the portal content will be included in the portal archive.

To export the portal, in `wpfas/modules/rest-service/servlet/src/java/oracle/webcenter/jaxrs/services/portal/controller/PortalsResource.java`, add a GET operation in the following format:

```
  @GET
  @Path("{portalId}/archive")
  public Response exportPortal(@PathParam("portalId") String portalId,
                  @DefaultValue(START_INDEX_DEFAULT)
                                                    @QueryParam("includePortalContent")
int includePortalContent);
```

The GET operation will call the API to perform the portal export. You can then download the PAR file to the local client. Response code 200 represents the successful export of a portal.

Importing 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 WebCenter Portal 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 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 Elasticsearch crawl to index the newly imported data.

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), and the security group of the target portal server is applied.

> **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 Migrating Discussions Resources for a Portal.

To find out how to import portal archives, see:

- Portal Import Prerequisites
- Importing a Portal from an Archive Using WebCenter Portal Administration
- Importing a Portal from an Archive Using WLST
- Importing a Portal Using REST APIs

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. If a shared identity store is not used, your system administrator must migrate users to the target. Refer to Back Up (Export) WebCenter Portal Schema Data and Restore (Import) WebCenter Portal Data.
- **Portals exist on the target** - Check whether any portals in the archive already exist on the target. If required, take existing portals offline during the import process, as described in Taking Any Portal Offline.
- **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.
- **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 Moving Connections Details from Staging to Production.
- **Archive version -** If you want to import portal archives from a WebCenter Portal 11*g* release, you must first upgrade to the current WebCenter Portal 12*c* release, re-create the portal export archive (.par file), and then import it.

For upgrade, see Understanding the Oracle WebCenter Upgrade Procedures Flow in *Upgrading Oracle WebCenter*.

Importing a Portal from an Archive Using WebCenter Portal Administration

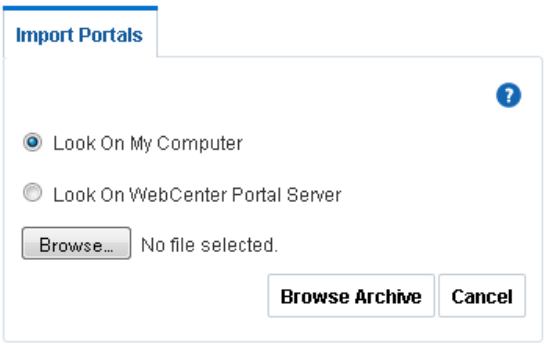
With `Portal Server-Manage All` or `Manage Configuration` permission, you can import portals from a portal archive through WebCenter Portal Administration.

To import one or more portals from a `.par` file:

1. Ensure that you meet the portal import prerequisites listed in Portal Import Prerequisites.
2. On the **Portals** administration page (see Accessing the Portals Page in WebCenter Portal Administration), click **Import** in the toolbar.

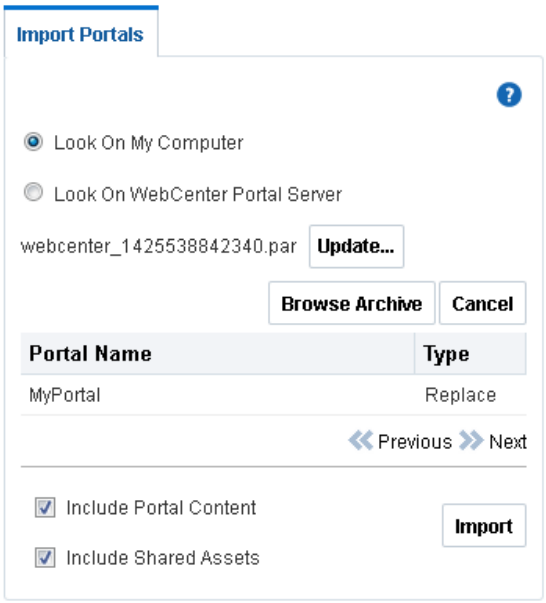
   The Import Portals dialog opens.

Figure 32-12 Importing Portals



3. 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.
4. Click **Browse Archive** to review the content available for import.

Figure 32-13 Importing Portals



The names of all the portals in the specified archive display in the table. The **Type** column indicates when there is a difference between the portals in the archive and those that 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.

5. Set import options as required.

Field	Description
Include Portal Content	(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, and the security group of the target portal server is applied. 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 Importing a Portal from an Archive Using WLST. If necessary, you can increase the upload setting, see Changing the Maximum File Upload Size.
Include Shared Assets	Select to import shared assets, like skin and page templates, used in the portal.

6. Click **Import**.

   - If you try to import portals that exist in the target WebCenter Portal application, the **Confirm Replace Portal** dialog displays. You must confirm whether you want to overwrite the existing portals.

     To delete existing portals and replace them with imported versions, click **Yes**. Click **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 Importing a Portal from an Archive Using WLST. If necessary, you can increase the upload setting. For details, see Changing the Maximum File Upload Size.

Note:

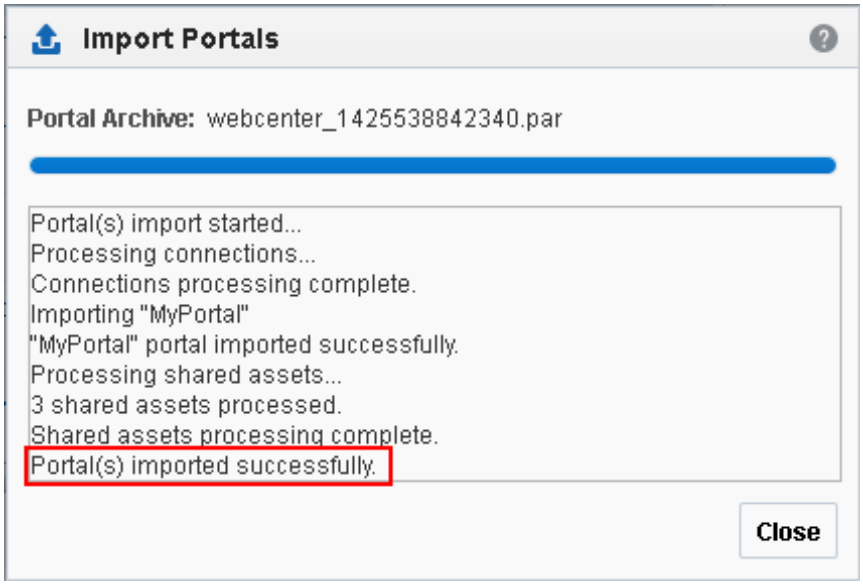
- If you are working with legacy portals with assets that store artifacts in MDS, Oracle recommends that you relocate all dependent artifacts from MDS to your content server. If you choose not to move artifacts stored in MDS and do not include MDS content within the asset archive, you can use MDS WLST commands `exportMetadata`/`importMetadata` to move the MDS content another time. For example:

```
exportMetadata(application='webcenter', server='WC_Portal',
 toLocation='/tmp/content',
 docs='/oracle/webcenter/siteresources/scopedMD/shared/**')

importMetadata(application='webcenter', server='WC_Portal',
 fromLocation='/tmp/content',
 docs='/oracle/webcenter/siteresources/scopedMD/shared/**')
```

7. In the information message, click **Yes** to confirm that you want to import the portals.

   An information message displays when all portals import successfully.

Figure 32-14 Portal Import Successful



8. Click **Close**.

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 Migrating Discussions Resources for a Portal.

- 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 Bringing Any Portal Back Online.

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, importConnections,
connPropertiesFile, importSharedAssets, server, applicationVersion])
```

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 or shared assets. 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 depend on your specific requirements. For command syntax, see `importWebCenterPortals` in *WebCenter WLST Command Reference Reference*.

For information on how to run WLST commands, see 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, and security, 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', importPortalContent=0, importActivities=0, overwrite=1,
savePortals=1)
```

Importing a Portal Using REST APIs

You can import portals into WebCenter Portal using REST API.

To import a portal using REST API, use the following URL format:

`http://`*`host:port`*`/rest/api/v1/portal/portals?utoken=<utoken_value>`

where *`host:port`* are the hostname and port number of the server into which you want to import the portal.

To import the portal, add the POST operation in the following format in the `wpfas/modules/rest-service/servlet/src/java/oracle/webcenter/jaxrs/services/portal/controller/PortalsResource.java`:

```
@POST
  @Consumes({MediaType.MULTIPART_FORM_DATA, MediaType.APPLICATION_OCTET_STREAM})
  @ResourceType("urn:oracle:webcenter:portal:portals")

  public Response importPortal(MultiPart multiPartData,
          @DefaultValue(START_INDEX_DEFAULT)

@QueryParam("includePortalContent") int includePortalContent);
```

For performing portal imports using the POST operation the content type must be specified as `multipart/form-data`. In a multipart format, each part is a contiguous portion of the object's data. You can upload each object part independently and in any order. If transmission of any part fails, you can retransmit that part without affecting other parts. The POST operation also requires the filename of the portal archive to be mapped to the `fileName` key.

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 `listWebCenterPortalArchive` in *WLST Command Reference for WebLogic Server*.

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

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, portal assets, and security information such as custom roles and member details. 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 WebCenter Portal Administration to export portal templates, the content stored in this folder is automatically included in the portal template archive for easy deployment to another target server.

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:

- Exporting Portal Templates
- Importing Portal Templates

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 WebCenter Portal Administration to export portal templates to an archive.

This section includes the following topics:

- Exporting Portal Templates to an Archive Using WebCenter Portal
- Exporting Portal Templates to an Archive Using WLST

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 information, see Exporting Portal Templates in *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 more information, see Exporting Online Portals to an Archive Using WebCenter Portal Administration.

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, exportConnections, server, applicationVersion])
```

The options that you set depends on your specific archive requirements. For command syntax, see `exportWebCenterPortalTemplates` in *WebCenter WLST Command Reference Reference*.

For information on how to run WLST commands, see 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')
```

Importing Portal Templates

Administrators can use the WLST command `importWebCenterPortals` to deploy one or more portal templates on a WebCenter Portal Server. Alternatively, administrators and application specialists can use WebCenter Portal 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 Publishing or Hiding Portal Templates in *Building Portals with Oracle WebCenter Portal*.

This section includes the following topics:

- Importing Portal Templates from an Archive Using WebCenter Portal
- Importing Portal Templates from an Archive Using WLST

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 Importing Portal Templates in *Building Portals with Oracle WebCenter Portal*.

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], [parentPortal],
[importCustomizations], [importPortalContent], [importSecurity], [importData],
[importActivities], [overwrite], [savePortals], [forceOffline],
```

```
[importLog], ]importConnections], [connPropertiesFile], [importSharedAssets],
[server], [applicationVersion])
```

The options that you set depend on your specific archive requirements. For command syntax, see importWebCenterPortals in *WebCenter WLST Command Reference Reference*.

For information on how to run WLST commands, see 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')
```

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 WebCenter Portal and administrators can perform the same tasks using WLST commands:

- Page templates
- Resource catalogs
- Skins
- Page styles
- Content Presenter display templates
- Visualizations
- Business objects
- Task flow styles
- Task flows
- Layout
- Data controls
- Data sources

When you download (or export) a WebCenter Portal asset, the asset details are saved to an export archive (`.aar` file). You can save the export archive to your local file system or a remote server file system using a filename of your choice. Artifacts, such as icons and images, used or referenced by assets are not included in the export or import archive unless they are stored in the portal's content folder on Content Server and the contents folder is in sync on the source and the target servers.

Devices and Device Groups

Administrators can export device groups and devices to a file (`.aar` 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:

- Exporting Assets, Devices, and Device Groups to an Archive
- Importing Assets from an Archive

Exporting Assets, Devices, and Device Groups to an Archive

This section describes the various ways you can create an asset, device, and device group archive. It includes the following topics:

- Exporting Assets to an Archive from WebCenter Portal
- Exporting Devices and Device Groups to an Archive
- Exporting an Asset, Device, or Device Group to an Archive Using WLST
- Exporting Assets Using REST API

See also, About Permissions Required to Import (or Export) Assets.

Exporting Assets to an Archive from WebCenter Portal

Administrators, application specialists, and portal managers can export assets from WebCenter Portal. For details, see Downloading an Asset in *Building Portals with Oracle WebCenter Portal*.

Exporting Devices and Device Groups to an Archive

This section includes the following topics:

- Exporting Devices and Device Groups Using WebCenter Portal
- Exporting Devices and Device Groups Using WLST

Exporting Devices and Device Groups Using WebCenter Portal

Administrators can export one or more devices and device groups to a file (`.par` file) from WebCenter Portal Administration. For details, see Managing Device and Device Group Lifecycles.

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 an export archive (`.aar` file):

```
exportWebCenterResource(appName, fileName, resourceType, [resourceGUID, resourceName,
spaceName, exportContentDirectory, server,
 applicationVersion])
```

For command syntax, see exportWebCenterResource in *WebCenter WLST Command Reference Reference*.

For information on how to run WLST commands, see 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.aar',
 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.aar',
 resourceType='device', resourceName='MyMobileDevice)
```

Exporting an Asset, Device, or Device Group to an Archive Using WLST

Administrators can use the WLST command `exportWebCenterResource` to export a single asset, device, or device group from WebCenter Portal:

```
exportWebCenterResource(appName, fileName, resourceType, [resourceGUID,
 resourceName, spaceName, exportContentDirectory, server, applicationVersion])
```

The options that you set depends on the asset, device, or device group you want to export. For command syntax, see exportWebCenterResource in *WebCenter WLST Command Reference Reference*.

For information on how to run WLST commands, see 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.aar`:

```
exportWebCenterResource(appName='webcenter',
 fileName='mySalesPageTemplateExport.aar', 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.aar`:

```
exportWebCenterResource(appName='webcenter', fileName='mySharedSkinExport.aar',
 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.aar',
 resourceType='skin', resourceName='MyCompanySkin)
```

Example 4- Exporting a device group

The following example exports a device group named "MyMobileDeviceGroup" from WebCenter Portal:

```
exportWebCenterResource(appName='webcenter', fileName='myDeviceGroupExport.aar',
 resourceType='deviceGroup', resourceName='MyMobileDeviceGroup)
```

Example 5- Exporting a device

The following example exports a device named "MyMobileDevice" from WebCenter Portal:

```
exportWebCenterResource(appName='webcenter', fileName='myDeviceExport.aar',
 resourceType='device', resourceName='MyMobileDevice)
```

Exporting Assets Using REST API

Oracle WebCenter Portal provides REST APIs to download a specific asset to an archive (.aar) from a portal or the shared assets area.

To export an asset using REST API, use the following URL format:

```
http://host:port/rest/api/v1/portal/typeOfAsset/assetId/archive?
utoken=utokenvalue
```

Where *`typeOfAsset`* refers to the asset you want to export, such as page templates, skins, visualization templates, or resource catalogs.

To export an asset, add the GET operation in the following format in `wpfas/modules/rest-service/servlet/src/java/oracle/webcenter/jaxrs/services/portal/controller/AssetTypeResource.java`

```
@GET
  @Path("{id}/archive")
  public Response exportPortal(@PathParam("id") String id);
```

Where, PathParam's *`id`* is the short ID of the asset to be exported.

Importing Assets from an Archive

You can only import an asset previously saved to a WebCenter Portal asset export archive (`.aar` file). For details, see Exporting Assets, Devices, and Device Groups 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:

- About Permissions Required to Import (or Export) Assets
- Importing Assets from an Archive using WebCenter Portal
- Importing Devices and Device Groups Using WebCenter Portal
- Importing Assets from an Archive using WLST
- Importing Assets Using REST API

About Permissions Required to Import (or Export) Assets

Table 32-3 describes the roles/permission required to import (or export) assets using the WebCenter Portal Administration.

> **Note:**
>
> If you want to import (or export) assets using WLST, you must also have the WebLogic Server `Monitor` role (or higher).

Table 32-3 Permissions Required to Import (or Export) Assets Using WebCenter Portal

Asset	Required WebCenter Portal Role or Permission	Description
Shared asset	• `Administrator` **OR**	• This role includes the required permissions for importing and exporting shared assets (`Create, Edit, Delete Assets` and `Manage Configuration`). See also, Managing Application Roles and Permissions.
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 WebCenter Portal Administration pages.
Portal asset	• `Portal Manager` **OR**	• This role includes the required permissions (`Create, Edit, Delete Assets` and `Manage Configuration`). See also, Managing Application Roles and Permissions.

Table 32-3 (Cont.) Permissions Required to Import (or Export) Assets Using WebCenter Portal

Asset	Required WebCenter Portal Role or Permission	Description
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.

Importing Assets from an Archive using WebCenter Portal

Administrators, application specialists, and portal managers can import assets from WebCenter Portal. For details, see Uploading an Asset in *Building Portals with Oracle WebCenter Portal*

Importing Devices and Device Groups Using WebCenter Portal

Administrators can import one or more devices and device groups from a file (`.par` file) using WebCenter Portal Administration. For details, see Managing Device and Device Group Lifecycles.

Importing Assets from an Archive using WLST

Administrators can use the WLST command `importWebCenterResource` to deploy a single asset, device, or device group to WebCenter Portal.

```
importWebCenterResource(appName, fileName, [resourceType, spaceName,
 overwriteContentDirectory, server, applicationVersion])
```

The options that you set depends on the asset, device, or device group you want to deploy. For command syntax, see importWebCenterResource in *WebCenter WLST Command Reference Reference*.

For information on how to run WLST commands, see 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.aar` in to the `Sales` portal:

```
importWebCenterResource(appName='webcenter',
 fileName='mySalesPageTemplateExport.aar', resourceType='pageTemplate',
 spaceName='SalesPortal')
```

Example 2 - Deploying a shared portal skin

The following example imports a shared portal skin archived in `mySharedSkinExport.aar`:

```
importWebCenterResource(appName='webcenter', fileName='mySharedSkinExport.aar',
 resourceType='skin')
```

Example 3 - Deploying a device group

The following example imports a device group exported to `myDeviceGroupExport.aar`:

```
importWebCenterResource(appName='webcenter', fileName='myDeviceGroupExport.aar',
 resourceType='deviceGroup')
```

Example 4 - Deploying a device

The following example imports a device archived in `myDeviceExport.aar`:

```
importWebCenterResource(appName='webcenter', fileName='myDeviceExport.aar',
 resourceType='device')
```

Importing Assets Using REST API

Oracle WebCenter Portal provides REST APIs to download a specific asset to an archive (`.aar` file) from a portal or the shared assets area.

To export an asset using REST API, use the following URL format:

```
http://host:port/rest/api/v1/portal/portals/portalShortId/typeOfAsset?
utoken=utokenvalue
```

Where *`typeOfAsset`* is the asset you want to export, such as page templates, skins, visualization templates, or resource catalogs, and *`portalShortId`* refers to the short ID of the portal into which the asset will be imported.

To import an asset into a portal or the shared assets area, add the POST operation in the following format in `wpfas/modules/rest-service/servlet/src/java/oracle/webcenter/jaxrs/services/portal/controller/PortalsReosurce.java`:

```
@POST
  @Consumes({MediaType.MULTIPART_FORM_DATA, MediaType.APPLICATION_OCTET_STREAM})
  @Path("{portalId}/<typeOfAsset>")
  @ResourceType("urn:oracle:webcenter:portal:<assetType>")
  public Response importPortal(@PathParam("portalId") String portalId,
                  MultiPart multiPartData);
```

Where, PathParam's *`portalId`* is the short ID of the portal into which asset will be imported, `multipartData` is the multipart data with the file to be consumed for upload.

For importing assets using the POST operation, the content type must be specified as `multipart/form-data`. In a multipart format, each part is a contiguous portion of the object's data. You can upload each object part independently and in any order. If transmission of any part fails, you can retransmit that part without affecting other parts. The POST operation also requires the filename of the asset archive to be mapped to the `fileName` key.

For importing a shared asset, you can also use the following URL format:

```
http://host:port/rest/api/v1/portal/typeOfAsset?utoken=utokenvalue
```

To import a shared asset, add the POST operation in the following format in `wpfas/modules/rest-service/servlet/src/java/oracle/webcenter/jaxrs/services/portal/controller/`*`AssetTypeResource`*`.java`:

```
POST
  @Consumes({MediaType.MULTIPART_FORM_DATA, MediaType.APPLICATION_OCTET_STREAM})
  @ResourceType("urn:oracle:webcenter:portal:<assetType>")
  public Response importPortal(MultiPart multiPartData)
```

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 Shared Libraries in *Developing for Oracle WebCenter Portal*.

If shared libraries are used by a portal, you can choose to push them to another instance while deploying or propagating the portal.

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 Understanding Connection Property Files.

This section includes the following topics:

- Exporting WebCenter Portal Connections Details to a File
- Importing New WebCenter Portal Connections from a File

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 `exportWebCenterPortalConnections` in *WebCenter WLST Command Reference Reference*.

For information on how to run WLST commands, see 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')
```

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 the WLST command `exportWebCenterPortalConnections`. To find out how, see 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 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 `importWebCenterPortalConnections` in *WebCenter WLST Command Reference Reference*.

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

Migrating Discussions Resources for a Portal

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. This includes migrating discussions resources if they are used in the portal.

Discussions:

- Exporting Portal Discussions to an Archive
- Importing Portal Discussions from an Archive

After importing one or more portals, consider initiating an search crawl to index the newly imported data.

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 Exploring the Settings Pages in WebCenter Portal 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.

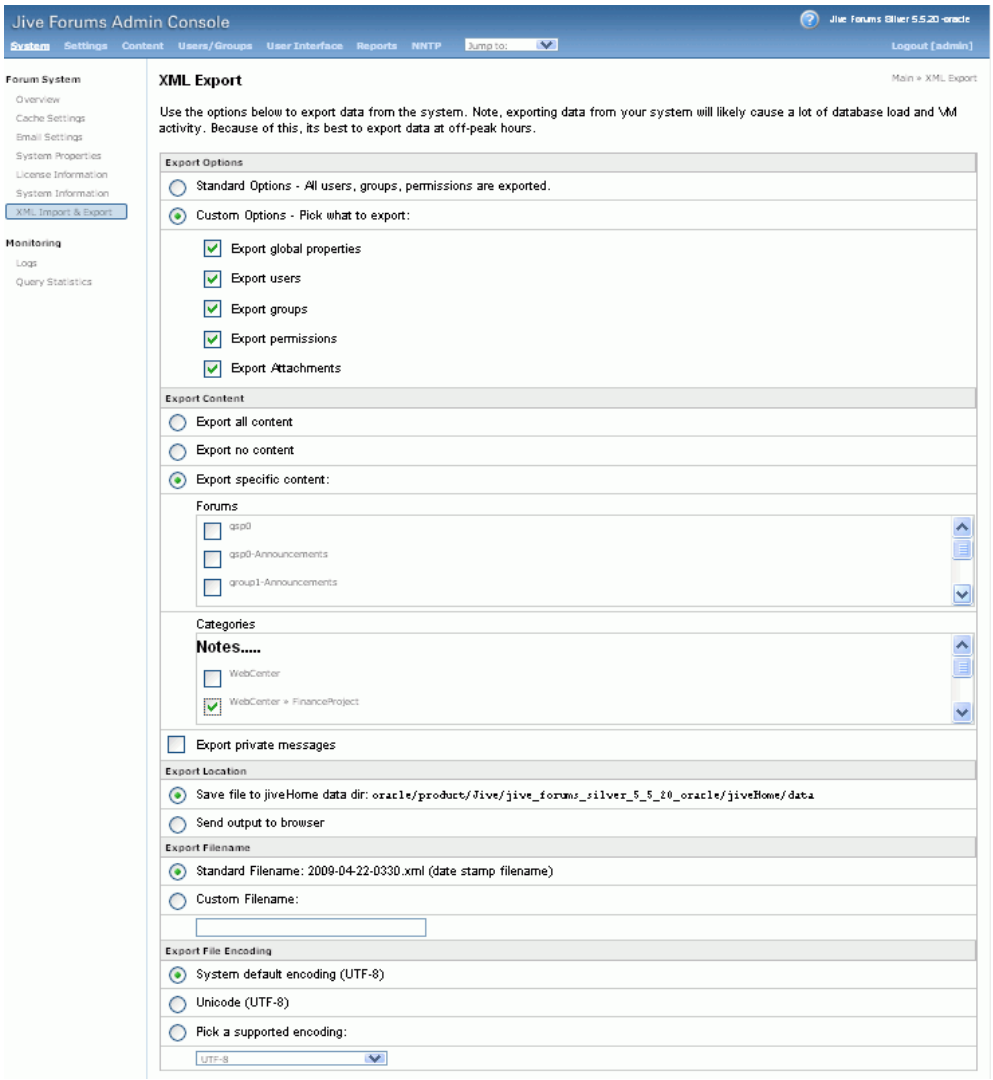
      WebCenter Portal's discussions server generates discussion category and forum IDs sequentially. If this ID exists on the target system, the imported forum (or category) will be assigned a new, unique ID, and therefore you must reconfigure the imported portal, to point to the new ID. For details, see 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 32-15):
   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 32-15 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 Importing a Portal from an Archive Using WebCenter Portal Administration.

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 Exporting and Importing Portal Archives or Directly Deploying Portals Using WLST.

> **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 Exploring the Settings Pages in WebCenter Portal 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**.

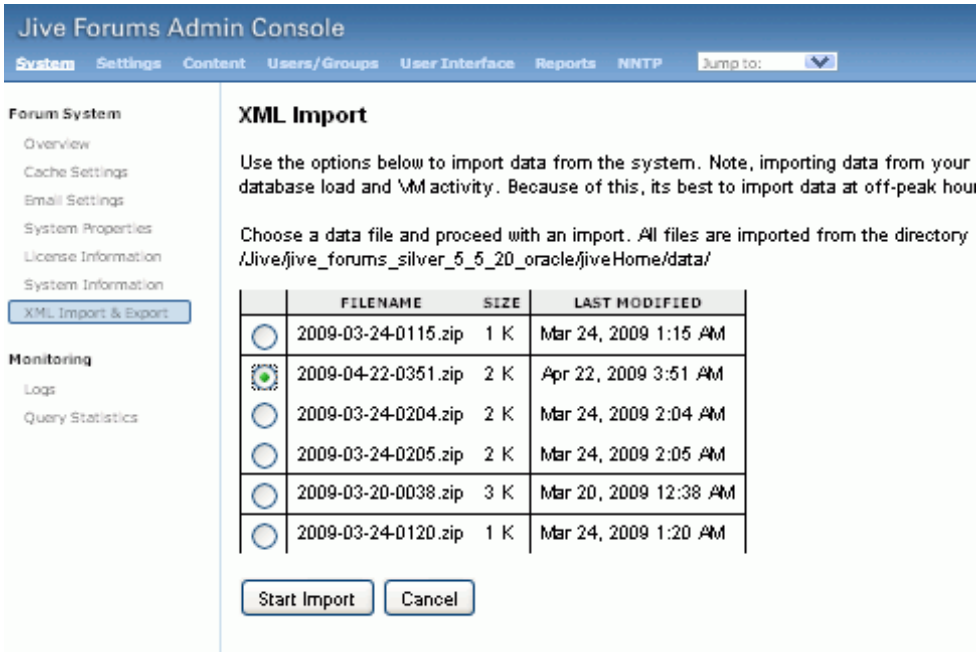
   e. Click **Administer Forums** (on the far right), and log into the Admin Console.

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 32-16).

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, 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 32-16 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.

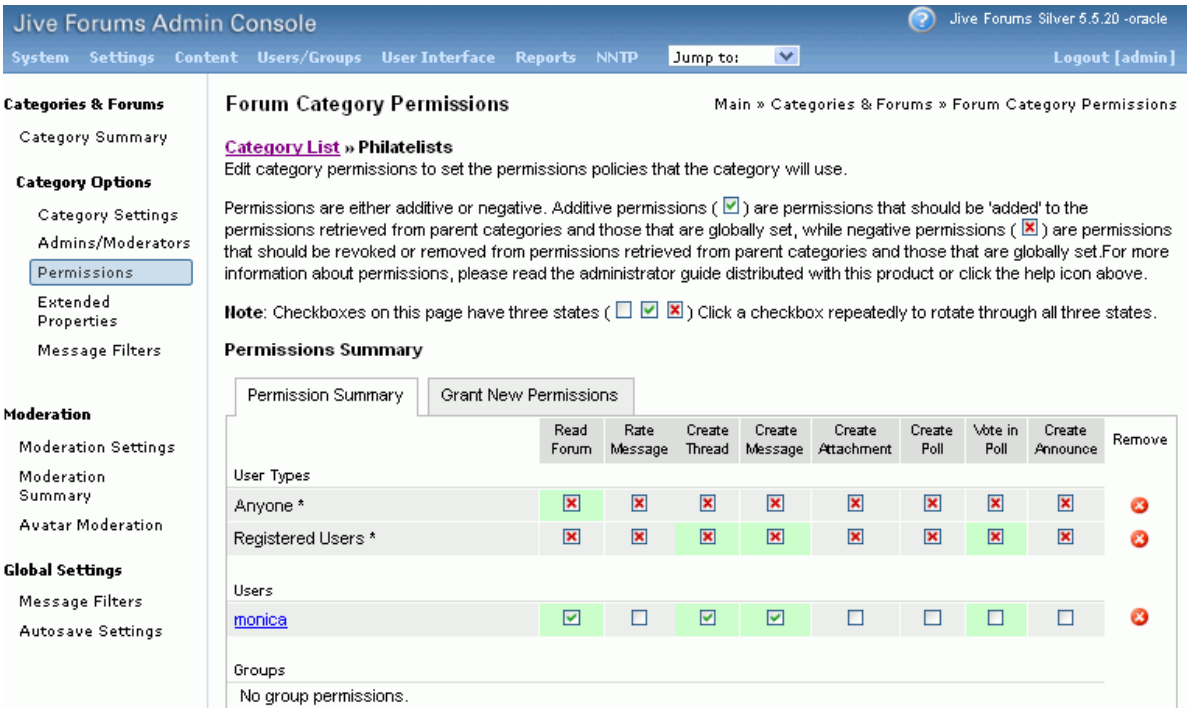
8. Select the root category for the target WebCenter Portal, and click **Move Categories**.

   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 32-17).

Figure 32-17 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 Exploring the Settings Pages in WebCenter Portal 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**.

Propagating and Redeploying Portals in Production

This section includes the following topics:

- Understanding Portal Propagation
- Propagating Portal Changes Using WebCenter Portal
- Propagating Portal Changes Using WLST
- Redeploying a Portal Using WebCenter Portal

Understanding Portal Propagation

Administrators can propagate portal changes made in staging to production if the stage and production environments are connected and kept "in sync". For example, you can propagate portal changes such as new pages and assets added or modified. Oracle strongly recommends that you *always* make changes in stage first and then push your portal changes to production using deployment or propagation. Propagation does not require the production server to be restarted or incur any downtime.

For lists of changes propagated from staging to production, see Table 32-4

Table 32-4 Portal Changes Propagated to Production

Portal Changes Propagated	Yes / No
Portal pages	Yes
Assets	Yes
Portlets	Yes
Portal folder content changes	Yes
Portal activity/usage data (activity streams, calendar events, feedback, list, links, message boards, people connections, profiles, surveys)	No
Portal security data excluding custom page security (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

Any structural changes to a portal require redeployment.

- Custom security can be set for portal pages. During portal propagation, custom page security changes are propagated. However, only portal-level security changes for existing roles (roles that are present on both the source and target servers) are propagated. If you created a new role and added new page permissions or added or removed members, changes are not propagated as the new role is not present on the target server. To migrate such changes, you must redeploy the portal.
- After deploying a portal if you enable a new tool, such as documents, or disable it on the source server and then propagate portal changes, tool-related changes are not reflected on the target server. You must redeploy your portal for the changes to take effect.

Propagating Portal Changes Using WebCenter Portal

To propagate changes made to a portal to the target server:

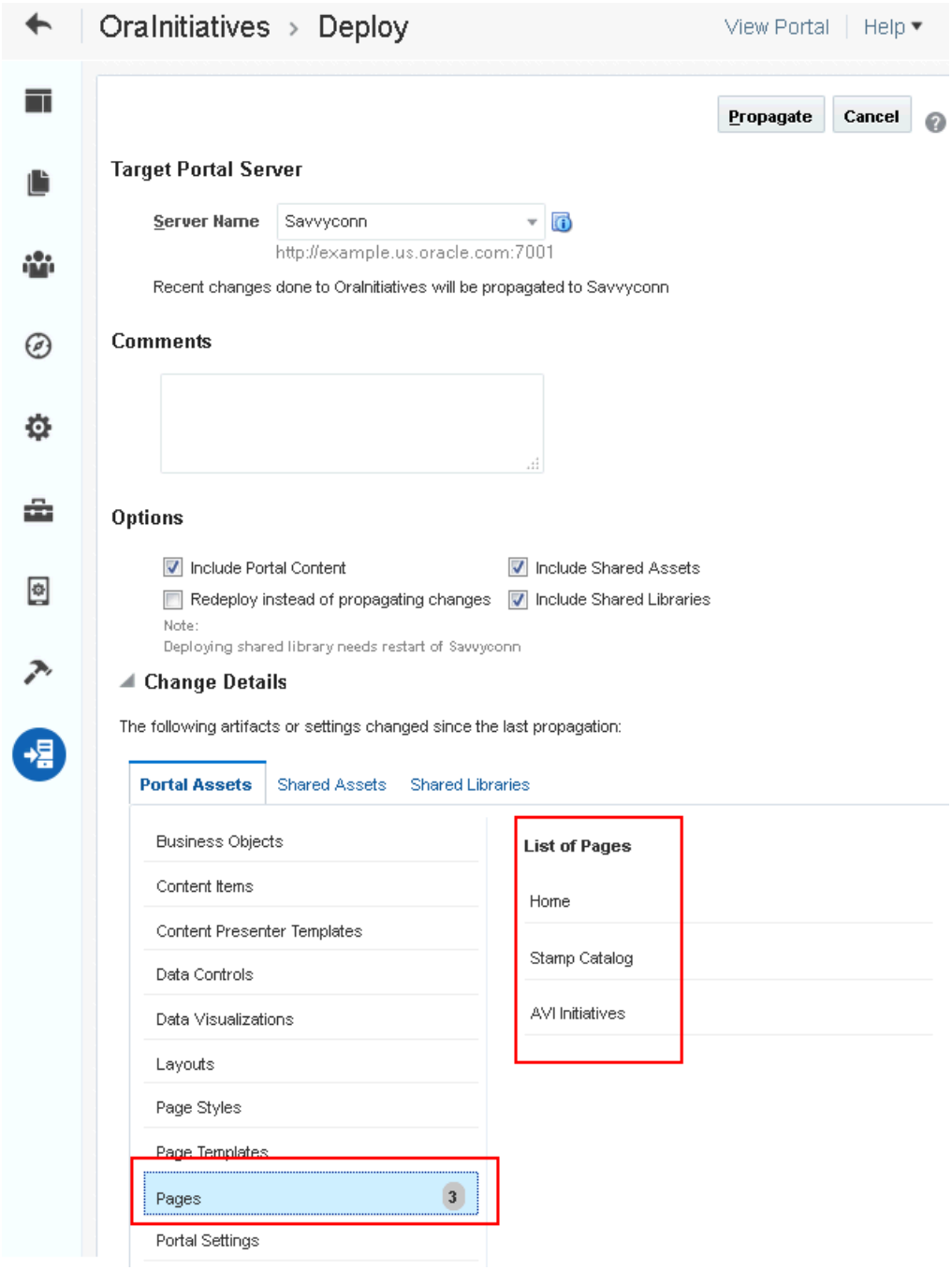
1. In WebCenter Portal, access portal administration as described in Accessing Portal Administration in *Building Portals with Oracle WebCenter Portal*.

2. Click the **Deploy** icon.

> **Note:**
>
> You will see the **Deploy** icon only if you are granted the application-level permission `Portal Server: Deploy`.

3. Under **Target Portal Server**, from the **Server Name** list, select the connection to be used for propagating portal changes. It must be the same connection that was used to deploy the portal.
4. In the **Comments** box, specify any comments related to portal propagation.
5. In the **Options** section, select the propagation options:
   - **Include Portal Content**: Select to specify that the portal content stored on Content Server must be propagated to the target server.
   - **Include Shared Assets**: Propagates the shared assets used by the portal. Clear the check box if you do not want to propagate shared assets.
   - **Include Shared Libraries**: Propagates the shared libraries used by the portal. Clear the check box if you do not want to propagate shared libraries. If you choose to propagate shared libraries, you must restart the target server after propagating portal changes for the shared library changes to be picked up.
6. Expand the **Changed Details** section to view the artifacts or settings that will be propagated.
   - **Portal Assets**: Lists the portal assets that have been added or updated since the last propagation. For example, Figure 32-18 the Pages category shows that three pages were added since the last propagation.
   - **Shared Assets**: Lists the shared assets used by the portal that were added or updated since the last propagation.
   - **Shared Libraries**: Lists the shared libraries used by the portal that were added or updated since the last propagation.

Figure 32-18 Propagating Portal Changes



7. Click **Propagate**.

   The Deploy Portals dialog displays the progress and status of portal propagation. While the portal is being propagated, you can choose to work on the portal if required.

8. Click **Close**.

9. If you propagated shared libraries, restart the target server for the shared libraries changes to take effect. For information, see Starting and Stopping Managed Servers for WebCenter Portal Application Deployments.

Propagating Portal Changes Using WLST

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 Directly Deploying Portals Using WLST.

To propagate metadata changes from staging to production:

- Run the WLST command `propagateWebCenterPortal` to propagate metadata for the portal.

  ```
  propagateWebCenterPortal(appName, portalName, targetConnectionName,
   [savePortal, propagateLog, propagateSharedAssets, propagatePortalContent,
  server, applicationVersion])
  ```

  The options that you set depends on your specific requirements. For command syntax, see propagateWebCenterPortal in *WebCenter WLST Command Reference Reference*.

  For information on how to run WLST commands, see 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 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 portal changes, including any changes to the portal content and shared assets used in the portal, 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',
propagateSharedAssets=1, propagatePortalContent=1,)
```

Redeploying a Portal Using WebCenter Portal

In WebCenter Portal, when you redeploy a portal, all information about the existing portal is deleted from the target server and a new portal is created.

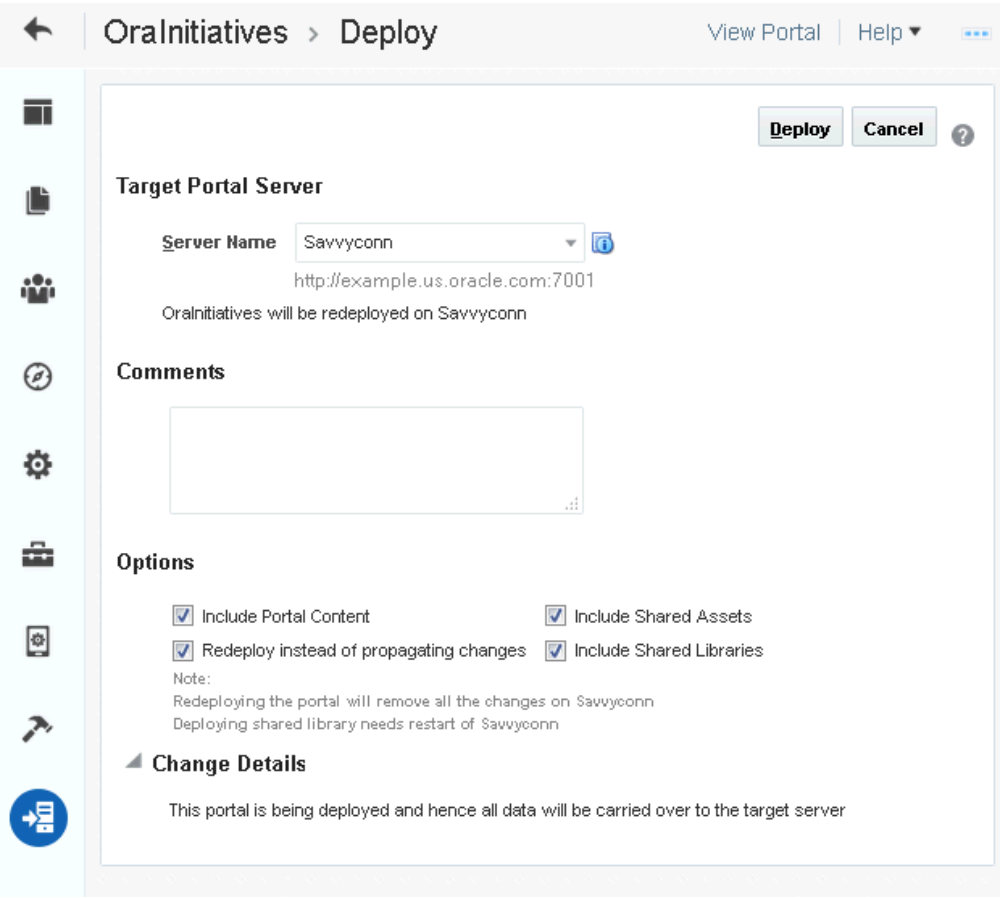
To redeploy a portal using WebCenter Portal:

1. In WebCenter Portal, access portal administration as described in Accessing Portal Administration in *Building Portals with Oracle WebCenter Portal*.
2. Click the **Deploy** icon.

> **Note:**
>
> You will see the **Deploy** icon only if you are granted the application-level permission `Portal Server: Deploy`.

3. In the **Comments** box, specify any comments related to portal redeployment.
4. Under **Target Portal Server**, from the **Server Name** list, select the connection that was used for deploying deploy the portal.
5. In the **Options** section, select the options:
   - **Redeploy instead of propagating changes**: Select to specify that the portal needs to be redeployed.
   - **Include Portal Content**: Select to specify that the portal content stored on Content Server must be included in portal redeployment.
   - **Include Shared Assets**: Deploys shared assets used by the portal. Clear the check box if you do not want to deploy shared assets.
   - **Include Shared Libraries**: Deploys the shared libraries used by the portal. Clear the check box if you do not want to deploy shared libraries. If you include shared libraries in portal deployment, you must restart the target server after redeploying the portal for shared library changes to be picked up.

Figure 32-19 Redeploying a Portal



6. Click **Deploy**.

   The Deploy Portals dialog displays the progress and status of portal deployment. While the portal is being redeployed, you can choose to work on the portal if required.

7. Click **Close**.

8. If you chose to redeploy shared libraries, restart the target server for the shared libraries changes to take effect. For information, see Starting and Stopping Managed Servers for WebCenter Portal Application Deployments.

33

Managing WebCenter Portal Backup, Recovery, and Cloning

This chapter describes techniques and tools for backing up and restoring WebCenter Portal.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

> **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 Permissions Required to Perform WebCenter Portal Lifecycle Operations.
>
> See also Understanding Administrative Operations, Roles, and Tools.

Topics:

- Understanding WebCenter Portal Back Up and Recovery
- Comparing Back up, Recovery, and Migration Tools for WebCenter Portal
- Backing Up Individual Portals
- Restoring Portals from a Backup
- Backing Up an Entire WebCenter Portal Installation
- Migrating Entire WebCenter Portal to Another Target
- Restoring an Entire WebCenter Portal Installation
- Using Scripts to Back Up and Restore WebCenter Portal
- Cloning a WebCenter Portal Environment

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 backup options. Administrators can back up:

- **One more more portals**

  WebCenter Portal provides export and import WLST commands for backing up and restoring individual portals. For details, see Backing Up Individual Portals and 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 Backing Up an Entire WebCenter Portal Installation and Using Scripts to Back Up and Restore WebCenter Portal.

> **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 Advanced Administration: Backup and Recovery in *Administering Oracle Fusion Middleware*.

Comparing Back up, Recovery, and Migration Tools for WebCenter Portal

Table 33-1 compares the various tools available to back up and restore WebCenter Portal or migrate WebCenter Portal to another target.

Table 33-1 Backup, Restore, and Migration Tools for WebCenter Portal

Category	Backup and Restore (Portals and Portal Templates)	Backup and Restore Scripts (Full WebCenter Portal Install)	Migration / Backup (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.

Table 33-1 (Cont.) Backup, Restore, and Migration Tools for WebCenter Portal

Category	Backup and Restore (Portals and Portal Templates)	Backup and Restore Scripts (Full WebCenter Portal Install)	Migration / Backup (WebCenter Portal Only)
When to use	Use to back up and restore portals 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	Content stored in the portal's content folder on Content Server, portal pages and assets, and portal data stored in persistence Portal security permissions and roles. For details, see: Understanding Portal Archives Deploying Portal Templates	MDS metadata for all tools and services, such as discussions, announcements, events, portlets, activities, tags, worklists, and so on. 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 and data stored in the WEBCENTER schema pages, application integration assets, lightweight content items, and tools and services, such as discussions, announcements, events, portlets, activities, and tags. 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, surveys, and tags.
What is not backed up / migrated	Any content outside of the portal's content folder on Content Server and any shared libraries used by the 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. Application-level settings stored in `adf-config.xml` (domain/MDS) Credentials (metadata/security/data/credentials). WebCenter Portal domain.

Table 33-1 (Cont.) Backup, Restore, and Migration Tools for WebCenter Portal

Category	Backup and Restore (Portals and Portal Templates)	Backup and Restore Scripts (Full WebCenter Portal Install)	Migration / Backup (WebCenter Portal Only)
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 content outside of the portal's content folder on Content Server, any shared library used by the portal, and 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 Advanced Administration: Backup and Recovery in *Administering Oracle Fusion Middleware*. 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 Moving from a Test to a Production Environment in *Administering Oracle Fusion Middleware*.

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.

Backing Up Individual Portals

The backup process for portals 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 Backing Up Portals Using WLST.

2. **Back up discussions and external data for the portal, if required**.

   See Backing Up Discussions and External Data for a Portal.

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 Using Scripts to Back Up and Restore WebCenter Portal.

See also, Restoring Portals from a Backup.

> **Note:**
>
> The simultaneous backup 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.

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 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, 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 `exportWebCenterPortals` in *WebCenter WLST Command Reference Reference*.

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

To restore the portal at a later date, see Restoring Portals from a Backup.

Backing Up Discussions and External 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 Exporting Portal Discussions to an Archive and Importing Portal Discussions from an Archive.

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.

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 Restoring Portals from an Archive Using WLST.

2. **Restore discussions data and external data for the portal, if required**.

   See Restoring Discussions and External Data for a Portal.

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 Using Scripts to Back Up and Restore WebCenter Portal.

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 Importing New WebCenter Portal Connections from a File.

- Take portals offline during portal restoration (`forceOffline=1`)

  Portal managers 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 depend on your specific requirements. For command syntax, see importWebCenterPortals in *WebCenter WLST Command Reference Reference*.

For information on how to run WLST commands, see 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. For information, see Restoring Discussions and External Data for a Portal.

Restoring Discussions and External Data for a Portal

Use the Discussions Server Admin Console to restore discussion data for a particular portal from a backup `.zip` file. For details, see Importing Portal Discussions from an Archive and Exporting Portal Discussions to an Archive.

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.

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 Back Up (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 analytics metadata.**

   See Backing Up and Restoring Analytics Metadata.

8. **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).

9. **Back up the WebLogic domain hosting WebCenter Portal**.

   See Backing Up and Restoring a WebCenter Portal Domain.

10. **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 Using Scripts to Back Up and Restore WebCenter Portal.

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, surveys, and tags.

This section includes the following topics:

- Prerequisites
- Back Up (Export) WebCenter Portal Schema Data
- Restore (Import) WebCenter Portal Data

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`

Back Up (Export) WebCenter Portal Schema Data

To back up `WEBCENTER` schema data, 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 command described in the example given below. For detailed `expdp` command information, see guide.

```
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, Restore (Import) WebCenter Portal Data.

Restore (Import) WebCenter Portal Data

To restore `WEBCENTER` schema data from a backup, use the appropriate utility for your database. 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.

   For example, to import WebCenter Portal schema data where source and target schema names match, run the following command:

```
DB_ORACLE_HOME/bin/impdp \"sys/password@serviceid as sysdba\"
 directory=mydmpdirectory dumpfile=webcenterportal.dmp SCHEMAS=tgtprefix_WEBCENTER
```

For example, to import WebCenter Portal schema data where source and target schema names are different, run the following command:

```
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.

For detailed `impdp` command information, see guide.

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

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.

Back Up (Export) All MDS Schema Data

To back up MDS data, 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 following command:

```
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.

For detailed expdp command information, see guide.

See also, Restore (Import) MDS Schema Data.

Restore (Import) MDS Schema Data

To restore `MDS` schema data from a backup, use the appropriate utility for your database. 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. For example, run the following command:

   ```
   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`.

For detailed `impdp` command information, see guide.

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 Advanced Administration: Backup and Recovery in *Administering Oracle Fusion Middleware*.

Optionally, you can back up the root folder for a WebCenter Portal instance 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 System Migration and Archiving in *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.

Backing up and Restoring Discussion Schema Data

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

Discussions and announcements store information in two database schemas:

- `DISCUSSIONS`: stores discussions and announcements data
- `DISCUSSIONS_CRAWLER`: enables Elastcsearch to crawl the discussions server.

This section includes the following topics:

- Prerequisites
- Back Up (Export) All Discussions Schema Data
- Restore (Import) Discussions Schema Data

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`

Back Up (Export) All Discussions Schema Data

To back up all discussions schema data, 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 following command:

> **Note:**
>
> This section describes how to export all discussions server data. If you want to export discussions for a single portal, see Backing Up Discussions and External Data for a Portal.

```
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.

For detailed `expdp` command information, see guide.

See also, Restore (Import) Discussions Schema Data.

Restore (Import) Discussions Schema Data

To restore discussions schema data from a backup, use the appropriate utility for your database. 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.

   For example, to import the discussions schema data where source and target schema names match, run the following command:

```
DB_ORACLE_HOME/bin/impdp \"sys/password@serviceid as sysdba\"
 directory=mydmpdirectory dumpfile=discussions.dmp
 SCHEMAS=tgtprefix_DISCUSSIONS,tgtprefix_DISCUSSIONS_CRAWLER
```

For example, to import the discussions schema data where source and target schema names are different, run the following command:

```
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.

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 PortalWebCenter Portal can store data is several other schemas:

- `ACTIVITIES` Stores analytics data

- PORTLET Stores portlet 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 this section.

  For detailed expdp and impdp command information, see 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)

The following example 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.

```
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)

This section describes 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.

   For example, to import schema data where source and target schema names match, run the following command:

   ```
   DB_ORACLE_HOME/bin/impdp \"sys/password@serviceid as sysdba\"
    directory=mydmpdirectory dumpfile=schemadump.dmp SCHEMAS=tgtprefix_SCHEMA_NAME
   ```

   For example, to import schema data where source and target schema names match, run the following command:

   ```
   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.

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 Advanced Administration: Backup and Recovery in *Administering Oracle Fusion Middleware*.

If you are using a different LDAP identity store, refer to the appropriate back up and recovery documentation for that product.

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. 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 Migrating Policies Manually in *Securing Applications with Oracle Platform Security Services*.

See also migrateSecurityStore in *WebCenter WLST Command Reference 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 Migrating Entire WebCenter Portal to Another Target.

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. 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 Migrating All Credentials with migrateSecurityStore in *Securing Applications with Oracle Platform Security Services*.

See also, migrateSecurityStore in *WebCenter WLST Command Reference Reference*.

Backing Up and Restoring a WebCenter Portal Domain

For information on how to back up and restore your domain configuration, see Advanced Administration: Backup and Recovery in *Administering Oracle Fusion Middleware*.

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. 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 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 Backing up and Restoring Other Schema Data (ACTIVITIES and PORTLET).

Backing Up (Exporting) Portlet Client Metadata

To export portlet client metadata and producer customizations and personalizations, for a single application, such as WebCenter Portal, 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 exportPortletClientMetadata in *WLST Command Reference for WebLogic Server*.

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

Restoring (Importing) Portlet Client Metadata

To import portlet client metadata and producer customizations and personalizations, for WebCenter Portal, 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, Backing Up (Exporting) Portlet Client Metadata.

For detailed syntax and examples, see the importPortletClientMetadata and exportPortletClientMetadata in *WebCenter WLST Command Reference Reference*.

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

Backing Up and Restoring Analytics Metadata

To back up the entire ACTIVITIES database schema, see Backing up and Restoring Other Schema Data (ACTIVITIES and PORTLET).

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 exportAuditConfig and importAuditConfig in *WLST Command Reference for WebLogic Server*.

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:

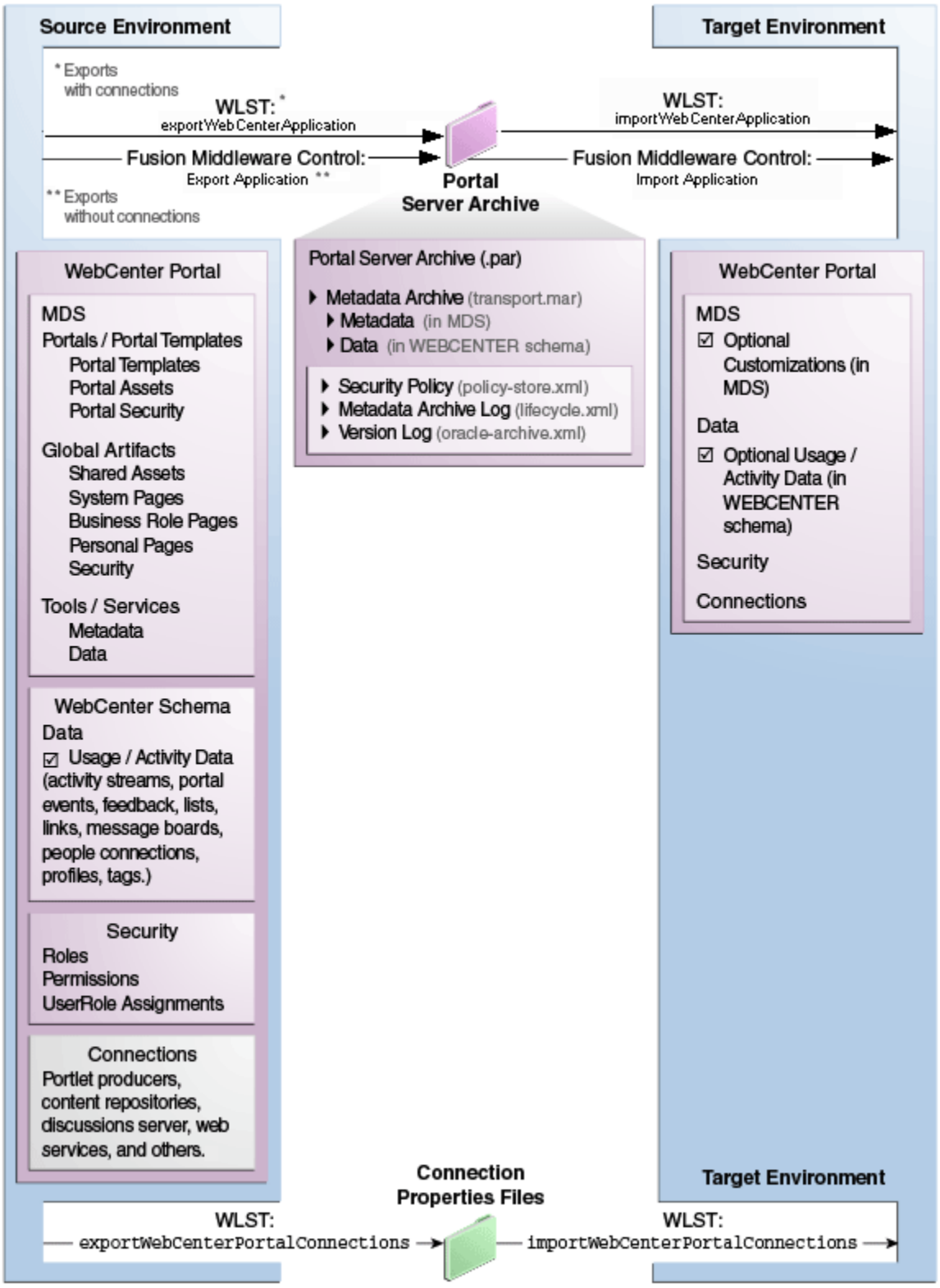
- Understanding Import and Export for WebCenter Portal
- Prerequisites for WebCenter Portal Export and Import
- Exporting WebCenter Portal to an Archive
- Importing a WebCenter Portal Archive

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 (`.par` file) using WLST commands or Fusion Middleware Control, as shown in Figure 33-1.

The WebCenter Portal export archive (`.par` file) contains several files, as listed in Figure 33-1.

Figure 33-1 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
- **Lightweight content** - Images from the all portals' content folder, styled text, and text.
- **Pages** - All pages, including system pages, business role pages, personal pages, and portal pages

In addition, the WebCenter Portal archive (`.par` 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 (`.par` 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, announcements, discussions, documents (on content server), instant messaging and presence (IMP), mail, calendar events, personalizations, and worklists. To learn how to backup or move data associated with these tools and services, see 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 Moving Connections Details from Staging to Production.

Information Always Exported and Imported

The following information is always included when you migrate WebCenter Portal to another target:

- Security Policy
  - policy-store.xml: Application roles and permissions and portal roles and permissions
  - User role assignments
- MDS – Shared / Portal Assets
  - Page Templates
  - Navigations
  - Resource Catalogs
  - Skins
  - Page Styles

  - Content Presenter Templates
  - Mashup Styles
  - Data Controls
  - Task Flows
- MDS – Tool/Service Data: Notes
- 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 – Portal Customizations: Portal administration settings
- MDS – User Customizations: Pages, task flows, and preferences
- WebCenter Portal Schema – Data
  - Activity Streams
  - Portal Events
  - Feedback
  - Links
  - Lists
  - Message Boards
  - Profiles
  - Tags
  - People Connections: Default settings for profiles, message boards, feedback, connections, activity streams; Activity stream task flow customizations

Information Never Exported and Imported

The following information is never included when you migrate WebCenter Portal to another target:

- External - Application Artifacts: icons and images
- External – Tool / Service Data
  - Documents (on content server)
  - Wikis and Blogs
  - Activity Graph
  - Analytics
  - Announcements
  - Discussions
  - IMP
  - Mail
  - Personal Events
  - Worklists
- Out-of-the -box: Portal templates and connections

> **Note:**
>
> Connections can be imported or exported based on options.

WebCenter Portal export and import can be performed using Fusion Middleware Control or WLST commands. For details, see:

- Prerequisites for WebCenter Portal Export and Import
- Exporting WebCenter Portal to an Archive
- Importing a WebCenter Portal Archive

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 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 Moving Connections Details from Staging to Production.

6. Ensure that the users in both the source and target environment are identical.

   > **Note:**
   >
   > 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, 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 Backing Up an Entire WebCenter Portal Installation.

8. Verify that the WebCenter Portal archive `.par` file that you want to import was exported from WebCenter Portal 12.2.1.

   You cannot import archives from earlier versions directly into WebCenter Portal 12.2.1. If necessary, you must upgrade your source environment to 12.2.1 before you create the export archive. For details, Upgrading Oracle WebCenter Portal in *Upgrading Oracle WebCenter*.

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 (`.par` 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 Understanding WebCenter Portal Back Up and Recovery.

This section includes the following:

- Exporting WebCenter Portal Using Fusion Middleware Control
- Exporting WebCenter Portal Using WLST

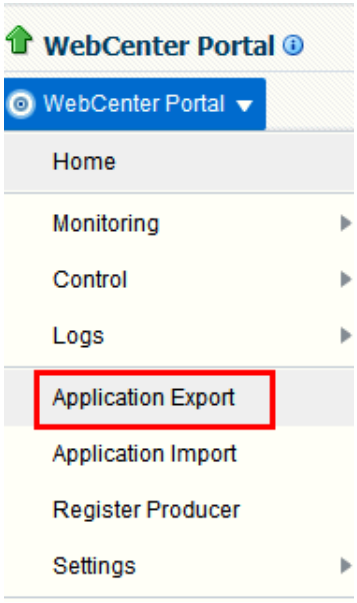
Exporting WebCenter Portal Using Fusion Middleware Control

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.
2. From the **WebCenter Portal** menu, select **Application Export**.

Figure 33-2 WebCenter Portal Menu - Application Export Option



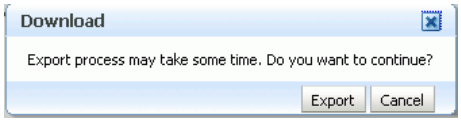
3. Change the **File Name** for the export archive or accept the default name.

   To ensure uniqueness, the default `.par` filename contains a unique ID—`webcenter_wholeapp_ts_`*`unique_ID`*`.par`.

Figure 33-3 Naming the Export Archive

4. Click **Export**.
5. In the Download dialog, click **Export** to confirm that you want to go ahead.

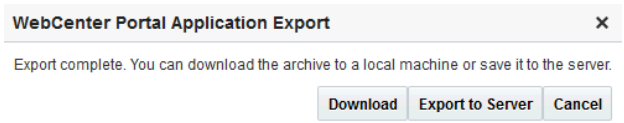
Figure 33-4 Downloading an Export Archive



Progress information is displayed during the export process. The application being exported cannot be accessed during export operations.

6. When the export process is complete, specify a location for the export archive (`.par`).

Figure 33-5 Saving an Export Archive



Select one of:

- **Download** - Saves the export PAR file to your local file system.

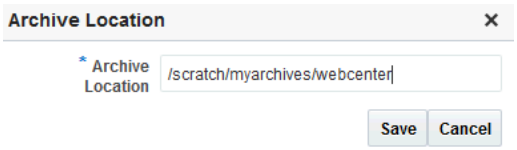
  Your browser downloads and saves the archive locally. The actual download location depends on your browser set up.

- **Export to Server** - Saves the export PAR file to a server location.

  When the Archive Location dialog displays, enter a suitable path for **Server Location**, for example, `/tmp`, and then click **Save**. The name of the PAR is not required here.

  Ensure that the server directory you specify has `write` permissions.

Figure 33-6 Archive Location



7. Click **Close** to dismiss the Export window.

The export archive (.PAR) is saved to the specified location.

Check the diagnostic log file, `WC_Portal-diagnostic.log`, for any warnings or errors reported during the export process. To view the log file, select the menu option **WebCenter Portal** then select, **Logs** and select,**View Log** , and then **Messages**. For details, see Viewing and Configuring WebCenter Portal Logs.

See also, Troubleshooting WebCenter Portal Import and Export.

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.par`. It also exports connections to the `connection.properties` file.

```
wls:/weblogic/serverConfig>exportWebCenterApplication(appName='webcenter',
fileName='myAppExport.par', connectionFileName='connection.properties')
```

The following example exports a test WebCenter Portal instance. The `.par` file is saved to the location from which you run the WLST command:

```
wls:/weblogic/serverConfig>exportWebCenterApplication(appName='webcenter',
fileName='myTestAppExport.par')
```

For command syntax and examples, see exportWebCenterApplication in *WebCenter WLST Command Reference Reference*.

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

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 Prerequisites for WebCenter Portal Export and Import.

This section includes the following:

- Importing WebCenter Portal Using Fusion Middleware Control
- Importing WebCenter Portal Using WLST
- Verifying WebCenter Portal After Import

Importing WebCenter Portal Using Fusion Middleware Control

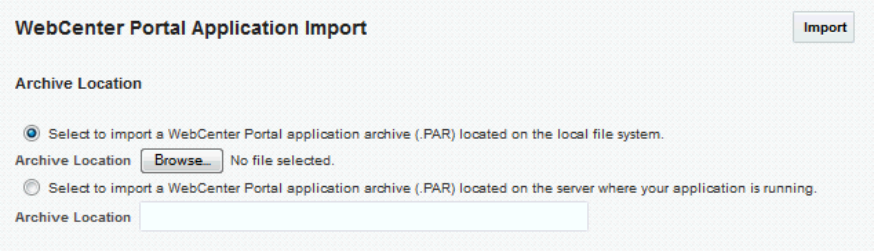
System administrators can import an entire WebCenter Portal instance using Fusion Middleware Control.

To import WebCenter Portal using Fusion Middleware Control:

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

   See Navigating to the Home Page for WebCenter Portal.
2. From the **WebCenter Portal** menu, select **Application Import**.
3. In the Application Import page (Figure 33-7), specify the location of your WebCenter Portal archive (`.par`).

Figure 33-7 Application Import Page



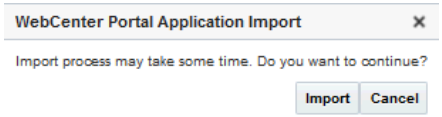
Select one of the following:

- **Archive Located on Local File System** - Enter the **Archive Location**. Alternatively, click **Browse** to locate the directory on the local file system where the `.par` 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 Importing Portals from an Archive for more information.

4. Click **Import**.
5. In the Application Import dialog (Figure 33-8), click **Import**.

Figure 33-8 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 Verifying WebCenter Portal After Import.

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 `importWebCenterApplication` in *WebCenter WLST Command Reference Reference* .

The following example imports WebCenter Portal from the export archive `myAppExport.par`:

```
wls:/weblogic/
serverConfig>importWebCenterApplication(appName='webcenter',fileName='myAppExport
.par')
```

For information on how to run WLST commands, see 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 Verifying WebCenter Portal After Import.

Verifying WebCenter Portal After Import

After importing WebCenter Portal from an archive you must:

1. Restart the managed server (`WC_Portal`) on which the newly imported WebCenter Portal instance is deployed.

   In a cluster environment, restart each managed server in the cluster. See also, 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 Portals and Portal Templates Not Available After Import.
3. Initiate the Elasticserch server crawler to index newly imported data.

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.

> **Note:**
>
> 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 analytics MDS metadata from a backup.**

  See Backing Up and Restoring Analytics 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 Using Scripts to Back Up and Restore WebCenter Portal.

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:

- Understanding Back Up and Restore Script Files
- Using Scripts to Back Up WebCenter Portal
- Restoring WebCenter Portal from Backups Using Scripts

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 33-2 describes the sample scripts and files provided for back up and recovery:

Table 33-2 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 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 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 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 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.

master_script.sh

`master_script.sh` 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
```

> **Note:**
>
> The `expdp` database command for individual schemas are described in 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 wlst_script.py.

The following is a sample `master_script.sh` script:

```
## 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.$$
```

wlst_script.py

The `wlst_script.sh` 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)`

> **Note:**
>
> If you want to back up or restore individual items, refer to the appropriate section in Backing Up an Entire WebCenter Portal Installation or Restoring an Entire WebCenter Portal Installation.

The following is a sample `wlst_script.py` script:

```
## 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
  disconnect()
  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
  disconnect()
  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)
```

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 33-3 lists and describes the input parameters in `backup.properties` and `restore.properties` files.

Table 33-3 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`
DB_CONNECT_PORTLET_SCHEMA	Connect string for the PORTLET database schema you want to export.	`mydbadmin/mypassword@db1234`

Table 33-3 (Cont.) User Defined Parameters for Back Up and Restore Scripts

Back up / Restore Parameter	Description	Example
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_Portal`
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_POLICY STORE	Name of a `jps-context` in the configuration file, where the source policy store is specified.	`mysourcePolicy`
TGT_JPS_CONTEXT_POLICYS TORE	Name of another `jps-context` in the configuration file, where the target policy store is specified.	`mytargetPolicy`

Table 33-3 (Cont.) User Defined Parameters for Back Up and Restore Scripts

Back up / Restore Parameter	Description	Example
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`

The following example shows a sample `backup.properties` file with sample values.

```
## 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_Portal

## 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/config/
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-system-
jazn-data.xml
CREDSTORE_FILE_NAME=/scratch/aime1/mywebcenterportal_scripts/backup/cwallet.sso
```

The following example shows a sample `restore.properties` file with sample values.

```
## 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_Portal

## 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/
config/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/
2013.05.30-08.39.2/backup-system-jazn-data.xml
CREDSTORE_FILE_NAME=/scratch/aime1/mywebcenterportal_scripts/mybackups/
2013.05.30-08.39.28/cwallet.sso
```

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 the sample code for `master_script.sh` from 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 the sample code for `wlst_script.py` from wlst_script.py, paste into a text editor, and save the file as `wlst_script.py` in the same directory.

4. Copy the sample code for `backup.properties` from backup.properties and restore.properties Files, 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.db"/>
</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.db"/>
</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 33-3 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.

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. Restore Database Schemas Manually (first time only).
3. Complete Prerequisite Tasks for Security Store Restore (first time only).
4. Set Restore Script Parameters.
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 the sample code for `restore.properties` from backup.properties and restore.properties Files, 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.

> **Note:**
>
> 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 33-3 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.

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 Additional Steps for Moving Oracle WebCenter Portal in *Administering Oracle Fusion Middleware*.

Part VII
Administering Multilanguage Portals

This part of *Administering Oracle WebCenter Portal* provides information about the language and translation topics for Oracle WebCenter Portal.

- Managing a Multilanguage Portal

34
Managing a Multilanguage Portal

Use the language support available in WebCenter Portal to manage translations at the application and portal level and for specific strings in a portal.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

> **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 WebCenter Portal Administration or a custom role that grants the following permission:
>
>   `Basic Services: Edit Page Access, Structure, and Content permission.`
>
> See also Understanding Administrative Operations, Roles, and Tools.

Topics:

- About Languages in WebCenter Portal
- Translating Strings for a Portal
- Translating Strings for Page Content
- Adding Support for a New Language to WebCenter Portal
- Translating Strings for Search Facets
- Translating Strings for Custom Search Attributes

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, including published text and images, documents, announcements, 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 in content publishing components can be translated, and your system administrator can display translated WebCenter Content items using

Content Presenter. Content added in announcements and discussion forums is generally displayed in the language used by the contributing user.

For information about providing localized content, see Translating Portals into Other Languages in *Building Portals with Oracle WebCenter Portal*.

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 34-1 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 34-1 Languages Available for WebCenter Portal

A to Ge	Gr to Ro	Ru to T
Arabic	Greek	Russian
Brazilian Portuguese	Hebrew	Simplified Chinese
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 34-1. 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.

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 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_Portal',toLocation='/tmp/metadata')
     ```

     This example exports all string files for WebCenter Portal (`webcenter`) on the `WC_Portal` 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_Portal',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 Running Oracle WebLogic Scripting Tool (WLST) Commands. See also exportMetadata in *WLST Command Reference for Infrastructure Components*.

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 Portal 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__7ff6:S
   ERVICEID:oracle.webcenter.page:OBJECTTYPE:page:OBJECTGUID::PAGES.:Page2.jspx.DISPLA
   Y_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_Portal',fromLocation='/tmp/
     metadata')
     ```

This example imports all string files from the `/tmp/metadata` folder to the `webcenter` application on the `WC_Portal` 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_Portal',fromLocation='
/tmp/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 Running Oracle WebLogic Scripting Tool (WLST) Commands. See also importMetadata in *WLST Command Reference for Infrastructure Components*.

8. Restart the `WC_Portal` managed server, and confirm that the changes you made appear in the UI.

Translating Strings for Page Content

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 a task flow on your portal page.

To translate the task flow header of a task flow added to your portal page:

1. Get the internal ID of the portal.
   a. In Oracle WebCenter Portal, navigate to the portal in which you want to implement translations.
   b. From the **Pages and Portals Actions** menu, select **Manage**, then **All Settings**.
   c. On the General page, from the **Internal ID** field, note down the internal ID of your portal. For example, `sc8f0a967_de9a_4f8e_aeaa_f10e5a100c94`.
2. Note down the Object GUID of the task flow that you want to translate:
   a. Open the portal page, which contains the task flow, in the edit mode.
   b. Select the task flow, then from the Actions menu, select **Display Options**.
   c. From the **Text** field, note down the value.

   For example, for the Message Board task flow the value might look like this:

   ```
   #{uib_o_w_s_r_DefaultGroupSpaceCatalog['WALL_MAINVIEW.TITLE']}
   ```

The EL expression shows the Object GUID of the task flow. So, for Message Board the Object GUID is `WALL_MAINVIEW.TITLE`.

3. Export the `scope-resource-bundle.xlf` file to a stage location by using the `exportMetadata` WLST command.

```
exportMetadata(application='webcenter',server='WC_Portal',toLocation='Stage-Folder',docs='/oracle/webcenter/translations/scopedMD/Portal-Internal-ID/**')
```

For example:

```
exportMetadata(application='webcenter',server='WC_Portal',toLocation='/home/oracle/testcases/translations/metadata',docs='/oracle/webcenter/translations/scopedMD/sc8f0a967_de9a_4f8e_aeaa_f10e5a100c94/**')
```

> **Note:**
>
> It is important that you export the translation files every time you want to make changes.

4. Back up the original `scope-resource-bundle.xlf` file. For example:

```
cd /home/oracle/testcases/translations/metadata/oracle/webcenter/translations/scopedMD/sc8f0a967_de9a_4f8e_aeaa_f10e5a100c94
cp scope-resource-bundle.xlf scope-resource-bundle.xlf_ori
```

5. Edit the `scope-resource-bundle.xlf` file and add a `trans-unit` block for the task flow you want to customize in the following format:

```
<trans-unit id="SCOPEGUID:<Portal-Internal-ID>:OBJECTGUID:object_GUID_taskflow">
<source>task_flow_title</source>
</trans-unit>
```

For example, you can add the following `trans-unit` block for the Message Board task flow:

```
<trans-unit
id="SCOPEGUID:sc8f0a967_de9a_4f8e_aeaa_f10e5a100c94:OBJECTGUID:WALL_MAINVIEW.TITLE">
<source>Message Board</source>
</trans-unit>
```

6. Create a language or locale-specific version of the `scope-resource-bundle.xlf` file for the desired language.

For example, for Arabic:

```
cp scope-resource-bundle.xlf scope-resource-bundle_ar.xlf
```

7. Edit the language or locale-specific version of the `scope-resource-bundle.xlf` file (for example, `scope-resource-bundle_ar.xlf` file) and make the appropriate translations.

For example, for the Message Board task flow ensure you have the following entry and translate it to the corresponding language:

```
<trans-unit id="SCOPEGUID:<Portal-Internal-
ID>:OBJECTGUID:WALL_MAINVIEW.TITLE">
<source>translated text for message board</source>
</trans-unit>
```

8. Import the modified files using the `importMetadata` WLST command.

```
importMetadata(application='webcenter',server='WC_Portal',fromLocati
on='Stage-Folder',docs='/oracle/webcenter/translations/scopedMD/
Portal-Internal-ID/scope-resource-bundle_lanugage_specific.xlf')
importMetadata(application='webcenter',server='WC_Portal',fromLocati
on='Stage-Folder',docs='/oracle/webcenter/translations/scopedMD/
Portal-Internal-ID/scope-resource-bundle.xlf')
```

For example:

```
importMetadata(application='webcenter',server='WC_Portal',fromLocati
on='/home/oracle/testcases/translations/metadata',docs='/oracle/
webcenter/translations/scopedMD/
sc8f0a967_de9a_4f8e_aeaa_f10e5a100c94/scope-resource-bundle_ar.xlf')
importMetadata(application='webcenter',server='WC_Portal',fromLocati
on='/home/oracle/testcases/translations/metadata',docs='/oracle/
webcenter/translations/scopedMD/
sc8f0a967_de9a_4f8e_aeaa_f10e5a100c94/scope-resource-bundle.xlf')
```

9. Update the task flow with the new value for the Text field:
   a. In WebCenter Portal, open the portal page, which contains the task flow, in the edit mode.
   b. Select the task flow, then from the Actions menu, select **Display Options**.
   c. In the Display Options dialog, update the value in the **Text** field.

      For example, for the Message Board task flow the value might look like this:

      ```
      #{uib_o_w_s_r_DefaultGroupSpaceCatalog['WALL_MAINVIEW.TITLE']}
      ```

      Replace this value with a value in following format:

      ```
      #{composerContext.inEditMode ?
      uib_o_w_s_r_DefaultGroupSpaceCatalog[‘OBJECTGUID'] : o_w_f_t_Transl
      ationsRBBean['task-flow trans-unit id']}
      ```

      For example:

      ```
      #{composerContext.inEditMode ?
      uib_o_w_s_r_DefaultGroupSpaceCatalog[' WALL_MAINVIEW.TITLE
      '] : o_w_f_t_TranslationsRBBean['SCOPEGUID:sc8f0a967_de9a_4f8e_aeaa
      _f10e5a100c94:OBJECTGUID:WALL_MAINVIEW.TITLE']}
      ```

   d. Click **OK**.
10. Save your portal page.

11. View your portal and test the translation by switching language to Arabic and back to English.

    You should be able to see the task flow title translated to the corresponding language you translated.

> **Note:**
>
> For task flows such as the RSS task flow, if you want to have several task flows of the same type you can append a number to have a different translation for each task flow. For example:
>
> ```
> <trans-unit
> id="SCOPEGUID:sc8f0a967_de9a_4f8e_aeaa_f10e5a100c94:OBJECTGUID:RSS
> _VIEWER.TITLE_1">
> <source>Oracle - Critical Patch Updates</source>
> </trans-unit>
> <trans-unit
> id="SCOPEGUID:sc8f0a967_de9a_4f8e_aeaa_f10e5a100c94:OBJECTGUID:RSS
> _VIEWER.TITLE_2">
> <source>OTN - Headlines</source>
> </trans-unit>
> ```

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 Using Spaces Extension Samples white paper on the *Oracle WebCenter Portal White Papers and Technical Notes* page on Oracle Technology Network.

Translating Strings for Search Facets

If your business is supported in multilanguage, you can translate the newly created or existing search facet strings. The `displayName` attribute contains the key of the translation string for a custom facet and is specified in the `search-service-custom-facets.xml` file.

The following is the sample of the `search-service-custom-facets.xml` file.

```
<?xml version='1.0' encoding='UTF-8'?>
<search-service-custom-facets xmlns="http://xmlns.oracle.com/webcenter/search/
customfacets">
.........................
<custom-facet name="CustomFacet" dataType="keyword" displayNameKey="WC_CustomFacet"
              displayName="Custom Facet"
mappedSearchAttribute="custom_metadata_field"/>
</search-service-custom-facets>
```

where,

- `name` is the name of the custom facet. In this example, the name is *CustomFacet*.
- `displayNameKey` is the value of the custom facet. In this example, the string is *WC_CustomFacet*.
- `displayName` is the display name of the custom facet, which appears in the Search Setting page of a portal.
- `Type` is the type of data.

To translate the `displayName` string, you need to create a language specific version of the `scope-resource-bundle.xlf` file for the desired language and import the file.

To translate the strings for facets:

1. Export the resource bundle, `scope-resource-bundle.xlf`, by using the `exportMetadata` WLST command:

   ```
   exportMetadata(application='webcenter',server='WC_Portal',toLocation='foldername',docs='/oracle/webcenter/translations/scopedMD/HomePortal-ID/scope-resource-bundle.xlf)
   ```

   where,

   - `toLocation='`*foldername*: is the location where you want to export the resource bundle, `scope-resource-bundle.xlf`
   - *HomePortal-ID* is the ID of the Home portal. For example, *s8bba98ff_4cbb_40b8_beee_296c916a23ed*.

   For example:

   ```
   exportMetadata(application='webcenter',server='WC_Portal',toLocation='/tmp/mds_dump',docs='/oracle/webcenter/translations/scopedMD/s8bba98ff_4cbb_40b8_beee_296c916a23ed/scope-resource-bundle.xlf)
   ```

   > **Note:**
   >
   > It is important that you export the translation files every time you want to make changes.

2. Navigate to the folder into which you exported the string files.

   ```
   cd foldername; cd oracle/webcenter/translations/scopedMD/HomePortal-ID
   ```

   where,

   - *foldername* is the location of the `scope-resource-bundle.xlf` file.
   - *HomePortal-ID* is the ID of the Home portal.

3. Create a language or locale-specific version of the `scope-resource-bundle.xlf` file for the desired language:

   For example, French:

   ```
   cp scope-resource-bundle.xlf scope-resource-bundle_fr.xlf
   ```

4. Edit the `scope-resource-bundle_fr.xlf` file and add a `trans-unit` block for the string that you want to translate in the following format:

```
<trans-unit id="SCOPEGUID:HomePortal-
ID:SERVICEID:oracle.webcenter.search:OBJECTTYPE:searchFacet:OBJECTGUID:CustomFacet:
OBJECTKEY:WC_CustomFacet">
<source>French Custom Facet</source>
</trans-unit>
```

where,

- SCOPEGUID is the ID of the Home portal. For example, *s8bba98ff_4cbb_40b8_beee_296c916a23ed*
- SERVICEID is oracle.webcenter.search.
- OBJECTTYPE is `searchFacet`
- OBJECTGUID is the name of the facet in the `search-service-custom-facets.xml` file. In this example it is *CustomFacet*.
- OBJECTKEY is the value of `displayName` field in the `search-service-custom-facets.xml` that corresponds to the `displayNameKey`, `WC_CustomFacet`.
- *Translated string* is the translated `displayName` string of the facet. In this example the translated string is `French Custom Facet`.

5. Import the modified files using the `importMetadata` WLST command.

```
importMetadata(application='webcenter', server='WC_Portal',
fromLocation='foldername',
docs='/oracle/webcenter/translations/scopedMD/HomePortal-ID/scope-resource-
bundle_fr.xlf')
```

where,

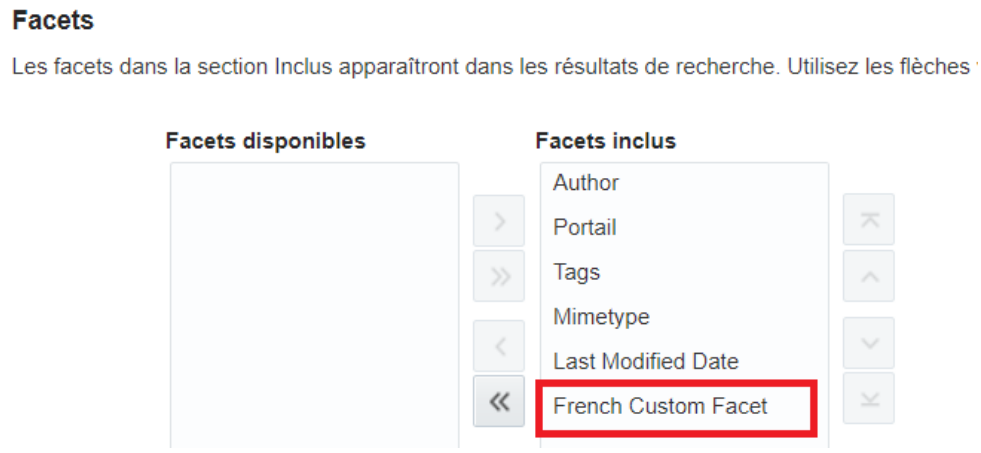
- `fromLocation='foldername`: is the location of the resource bundle, `scope-resource-bundle.xlf` from where you want to import.
- *HomePortal-ID* is the ID of the Home portal. For example, *s8bba98ff_4cbb_40b8_beee_296c916a23ed*.

For example:

```
importMetadata(application='webcenter', server='WC_Portal', fromLocation='/tmp/
mds_dump',
docs='/oracle/webcenter/translations/scopedMD/
s8bba98ff_4cbb_40b8_beee_296c916a23ed/scope-resource-bundle_fr.xlf')
```

6. Log in to Oracle WebCenter Portal and change the language to corresponding language you translated. In this example, the translated string is in French.
7. Open the Search Settings page in Tools and Services and view the translated value for the facet.

   You should be able to see the facet translated to the corresponding language you translated.



Translating Strings for Custom Search Attributes

If your business is supported in multilanguage, you can translate the newly created or existing search attribute strings. The `displayName` attribute contains the key of the translation string for a search attribute and is specified in the `search-service-attributes.xml` metadata file.

The following is the sample of the `search-service-attributes.xml` metadata file.

```
<service id="oracle.webcenter.doclib">
.......................
.......................
<attribute name="custom_metadata_field" displayNameKey="WC_CustomMetadataField"
           displayName="Custom Metadata Field" type="keyword"
backendAttribute="xCustomMetadataField"/>
</service
```

where,

- `name` is the name of the custom attribute field. In this example, the name is *custom_metadata_field*.
- `displayNameKey` is the value of the custom attribute field. In this example, the string is *WC_CustomMetadataField*.
- `displayName` is the display name of the custom attribute, which appears in the Search Setting page of a portal and contains the string for the translation.
- `Type` is the type of data.

To translate the `displayName` string, you need to create a language specific version of the `scope-resource-bundle.xlf` file for the desired language and import the file.

To translate the strings for custom search attributes:

1. Export the resource bundle, `scope-resource-bundle.xlf`, by using the `exportMetadata` WLST command:

   ```
   exportMetadata(application='webcenter',server='WC_Portal',toLocation='foldername',docs='/oracle/webcenter/translations/scopedMD/HomePortal-ID/scope-resource-bundle.xlf)
   ```

   where,

- `toLocation='foldername`: is the location where you want to export the resource bundle, `scope-resource-bundle.xlf`
- *HomePortal-ID* is the ID of the Home portal. For example, *s8bba98ff_4cbb_40b8_beee_296c916a23ed*.

For example:

```
exportMetadata(application='webcenter',server='WC_Portal',toLocation='/tmp/
mds_dump',docs='/oracle/webcenter/translations/scopedMD/
s8bba98ff_4cbb_40b8_beee_296c916a23ed/scope-resource-bundle.xlf)
```

> **Note:**
>
> It is important that you export the translation files every time you want to make changes.

2. Navigate to the folder into which you exported the string files.

```
cd foldername; cd oracle/webcenter/translations/scopedMD/HomePortal-ID
```

where,

- *foldername* is the location of the `scope-resource-bundle.xlf` file.
- *HomePortal-ID* is the ID of the Home portal.

3. Create a language or locale-specific version of the `scope-resource-bundle.xlf` file for the desired language:

For example, French:

```
cp scope-resource-bundle.xlf scope-resource-bundle_fr.xlf
```

4. Edit the `scope-resource-bundle_fr.xlf` file and add a `trans-unit` block for the string that you want to translate in the following format:

```
<trans-unit id="SCOPEGUID:HomePortal-
ID:SERVICEID:oracle.webcenter.search:OBJECTTYPE:searchAttribute:OBJECTGUID:custom_m
etadata_field:OBJECTKEY:WC_CustomMetadataField"
<source>French Custom Metadata Field</source>
</trans-unit>
```

where,

- SCOPEGUID is the ID of the Home portal. For example, *s8bba98ff_4cbb_40b8_beee_296c916a23ed*
- SERVICEID is oracle.webcenter.search.
- OBJECTTYPE is `searchAttribute`
- OBJECTGUID is the name of the custom attribute in the `search-service-attributes.xml` file. In this example it is *custom_metadata_field*.
- OBJECTKEY is the value of `displayName` field in the `search-service-attributes.xml` . *WC_CustomMetadataField*
- *Translated string* is the translated `displayName` string of the search attribute. In this example, the string is `French Custom Metadata Field`.

5. Import the modified files using the `importMetadata` WLST command.

```
importMetadata(application='webcenter', server='WC_Portal',
fromLocation='foldername',
docs='/oracle/webcenter/translations/scopedMD/HomePortal-ID/scope-resource-
bundle_fr.xlf')
```

where,

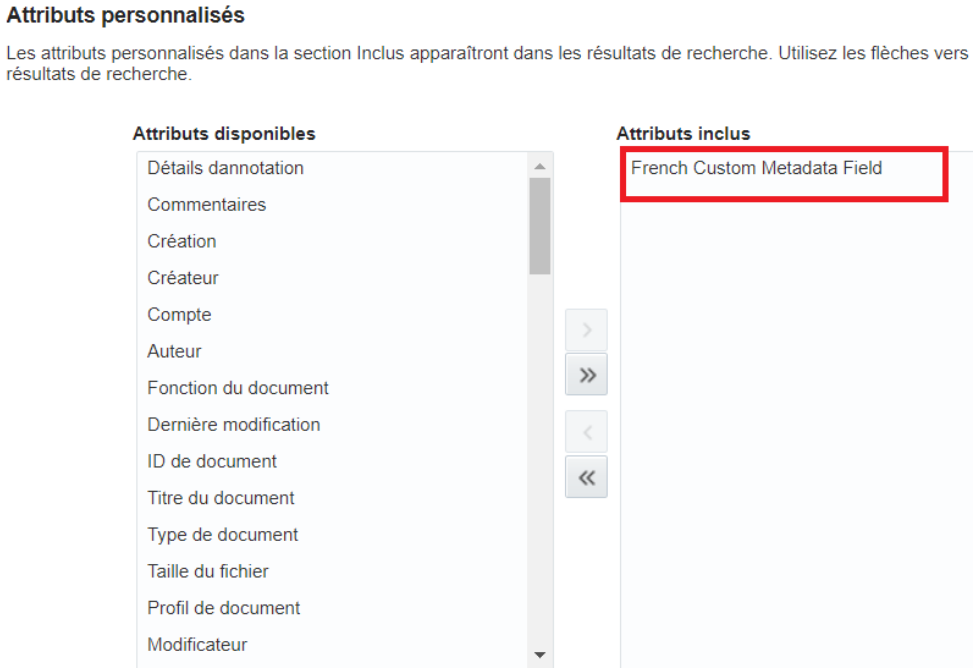
- `fromLocation='foldername`: is the location of the resource bundle, `scope-resource-bundle.xlf` from where you want to import.
- *HomePortal-ID* is the ID of the Home portal.

For example:

```
importMetadata(application='webcenter', server='WC_Portal',
fromLocation='/tmp/mds_dump',
docs='/oracle/webcenter/translations/scopedMD/
s8bba98ff_4cbb_40b8_beee_296c916a23ed/scope-resource-bundle_fr.xlf')
```

6. Log in to Oracle WebCenter Portal and change the language to corresponding language you translated. In this example, the translated string is in French.
7. Open the Search Settings page in Tools and Services and view the translated value for the custom attribute.

   You should be able to see the attribute translated to the corresponding language you translated.



Part VIII

Administering Portals in WebCenter Portal

This part of *Administering Oracle WebCenter Portal* describes how to administer global settings for WebCenter Portal users on the pages in WebCenter Portal Administration.

- Exploring the Settings Pages in WebCenter Portal Administration
- Exploring the Portals Page in WebCenter Portal Administration
- Configuring Global Defaults Across Portals
- Managing Security Across Portals
- Working with Global Attributes Across Portals
- Customizing System Pages
- Managing Business Role Pages
- Managing Personal Pages
- Administering Device Settings
- Customizing Task Flows
- Analyzing Portal Usage

35

Exploring the Settings Pages in WebCenter Portal Administration

Use the **Settings** pages in WebCenter Portal Administration to set application-level properties for WebCenter Portal.

> **Permissions:**
>
> To perform the tasks in this chapter, you must have the WebCenter Portal `Administrator` role or a custom role that grants the permissions required by the specific tasks that you want to perform.

> **Note:**
>
> - 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.

Topics:

- Working with WebCenter Portal Administration Settings
- Accessing the Settings Pages in WebCenter Portal Administration

Working with WebCenter Portal Administration Settings

The **Settings** pages in WebCenter Portal Administration enable system administrators to perform the tasks described in Table 35-1. The table also lists the permissions required to perform the various tasks.

Figure 35-1 WebCenter Portal Administration Settings

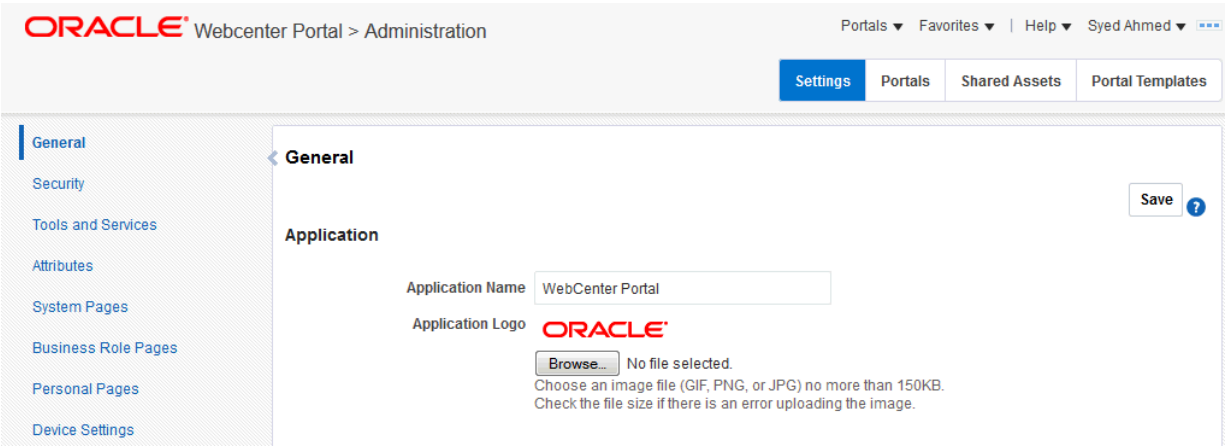


Table 35-1 WebCenter Portal Administration Settings Pages

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
	• Session timeout options and setting
	• self-registration options
	For more information, see Configuring Global Defaults Across Portals .
		`Portal Server: 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 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 Managing Tools and Services.
		`Portal Server: Manage All`
		or
		`Portal Server: Manage Configuration`
		Note: Some tools and services may require additional permissions. For example: people connections, portlet producers, and external applications require the WebCenter Portal `Administrator` role and the WebLogic Server `Admin` role

Table 35-1 (Cont.) WebCenter Portal Administration Settings Pages

Page	Description	Required Permission

Attributes	Use this page to manage settings for attributes in WebCenter Portal.
	For more information, see Working with Global Attributes Across Portals.
		`Portal Server: Manage All`
		or
		`Portal Server: Manage Configuration`

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 Customizing System Pages.
		`Portal Server: Manage All`
		or
		`Portal Server: Manage Configuration`
		or
		`Pages: 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 Managing Business Role Pages.
		`Portal Server: Manage All`
		or
		`Portal Server: Manage Configuration`
		or
		`Pages: 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 Managing Personal Pages.
		`Portal Server: Manage All`
		or
		`Portal Server: Manage Configuration`
		or
		`Pages: Create, Edit, and Delete Pages`

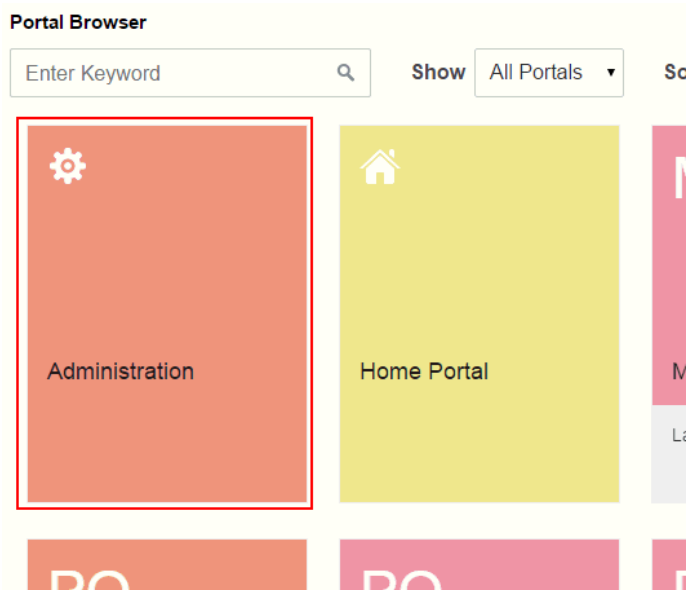
Device Settings	Use this page to create and manage device groups and devices for WebCenter Portal.You can create a device group, associate various devices with it, and specify the assets, such as the skin and page template, to be used for the device group.
	For more information, see Administering Device Settings.
		`Portal Server: Manage All`
		or
		`Portal Server: Manage Configuration`

Accessing the Settings Pages in WebCenter Portal Administration

Open the **Settings** pages in WebCenter Portal Administration in the following ways:

- From the **Portals** menu, select **Administration**, then click **Settings**.
- In the portal browser, click the **Administration** tile (Figure 35-2), then click **Settings**.

Figure 35-2 WebCenter Portal Administration Access



- Enter the following URL in your browser to navigate directly to the **Settings** pages:

  `http://host:port/webcenter/portal/admin/settings`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

36

Exploring the Portals Page in WebCenter Portal Administration

Use the **Portals** page in WebCenter Portal Administration to edit and administer all portals. The tasks available to you are dependent on your role or permissions.

> **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:
>
> - `Portals: Manage Security and Configuration`
>
> If you are a portal manager (or have the `Administration: Manage Security and Configuration` or `Administration: Manage Configuration` permission in a portal), you can perform these tasks on that portal alone, as described in Administering a Portal in *Building Portals with Oracle WebCenter Portal*.
>
> For more information about permissions, see About Application Roles and Permissions.

Topics:

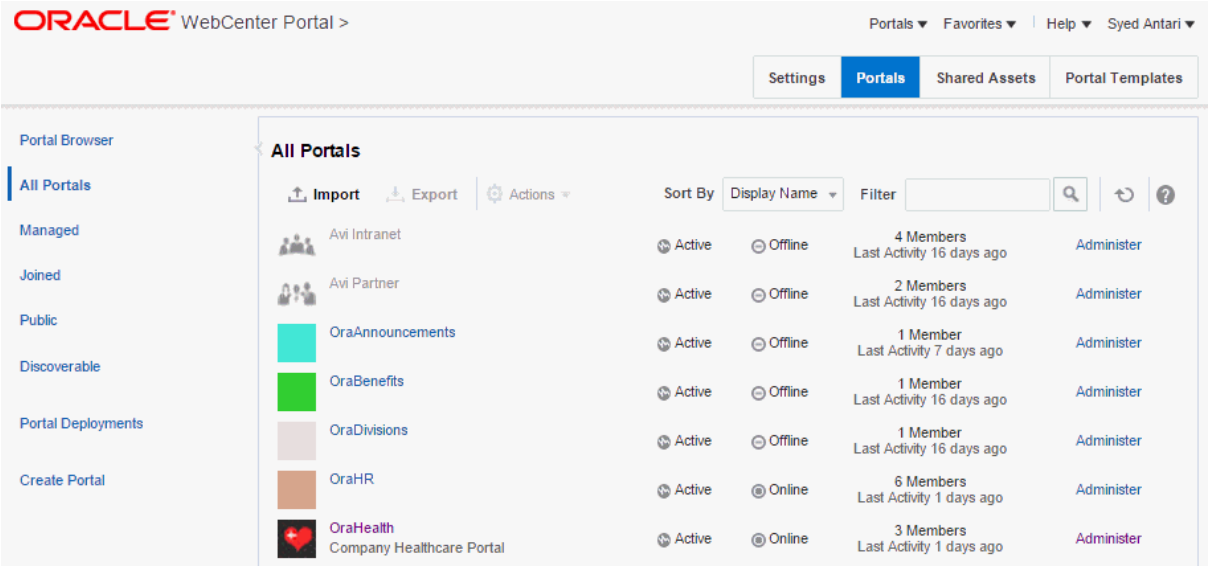
- About the Portals Page in WebCenter Portal Administration
- Accessing the Portals Page in WebCenter Portal Administration
- Sorting the Portals Listing
- Creating a Portal
- Exporting and Importing a Portal
- Viewing Information About Any Portal
- Sharing the Link to a Portal
- Closing Any Portal
- Reactivating Any Portal
- Taking Any Portal Offline
- Bringing Any Portal Back Online
- Deleting a Portal

About the Portals Page in WebCenter Portal Administration

The **Portals** page in WebCenter Portal Administration (Figure 36-1) provides access to editing and administering all portals in WebCenter Portal, including exporting and importing portals.

If granted appropriate permissions, users can use this page to 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 Administering a Portal in *Building Portals with Oracle WebCenter Portal*.

Figure 36-1 WebCenter Portal Administration - Portals Page

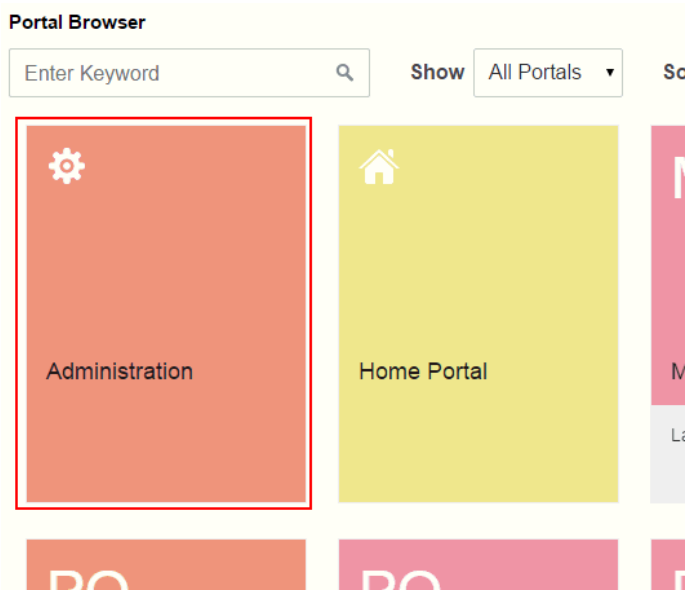


Accessing the Portals Page in WebCenter Portal Administration

To manage all portals in WebCenter Portal:

1. Open the **Portals** page in WebCenter Portal Administration in either of the following ways:
   - From the **Portals** menu, select **Administration**, then click **Portals**.
   - In the portal browser, click the **Administration** tile (Figure 36-2), then click **Portals**.

Figure 36-2 WebCenter Portal Administration Access



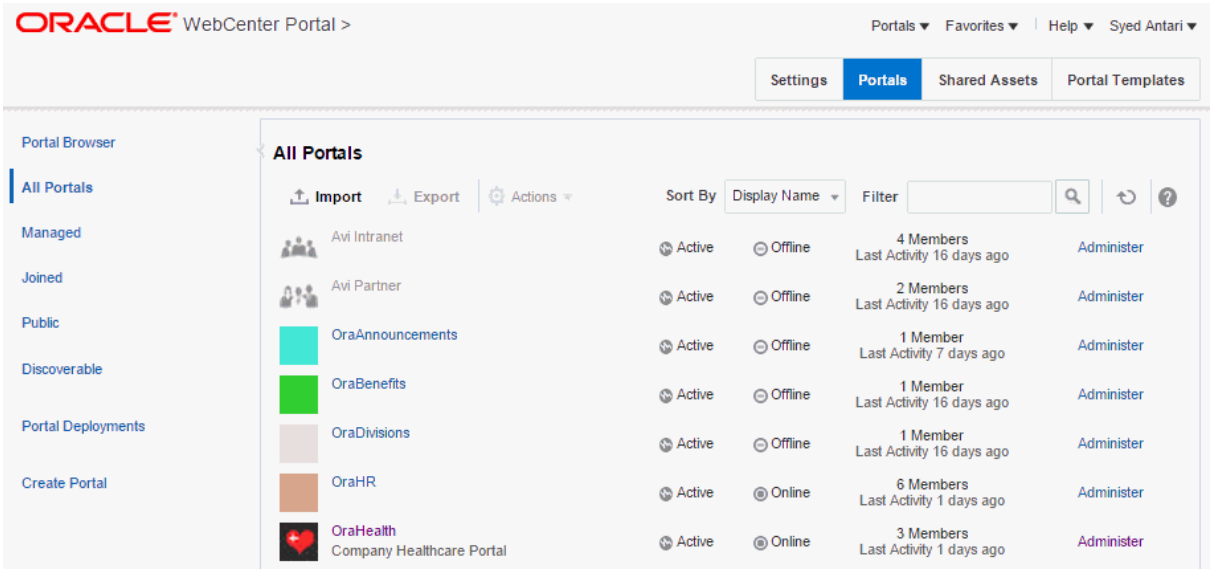
- Enter the following URL in your browser to navigate directly to the **Portals** page:

  `http://host:port/webcenter/portal/admin/portals`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

The **Portals** page displays (Figure 36-3).

Figure 36-3 WebCenter Portal Administration - Portals Page



2. On the **Portals** administration page, in the left pane, select:
   - **All Portals** to show all portals that are available to you, both public and private.

     > **Note:**
     >
     > Hidden portals can be seen on this page by users with the `Portal Server: Manage Configuration` or `Portals: Manage Security and Configuration`permission, such as a system administrator. While these users can manage the portal (change settings and membership), they cannot see the portal pages and content unless they are a portal member.

   - **Managed** to display portals for which you have portal manager 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.

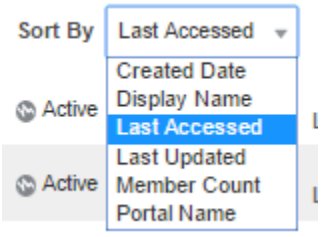
Sorting the Portals Listing

To sort the list of portals on the **Portals** page:

1. On the **Portals** administration page, click the **Sort By** selection list.

   > **Note:**
   >
   > When **All Portals** is selected in the left selection pane, you can sort by only **Display Name** and **Last Accessed**.

Figure 36-4 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.

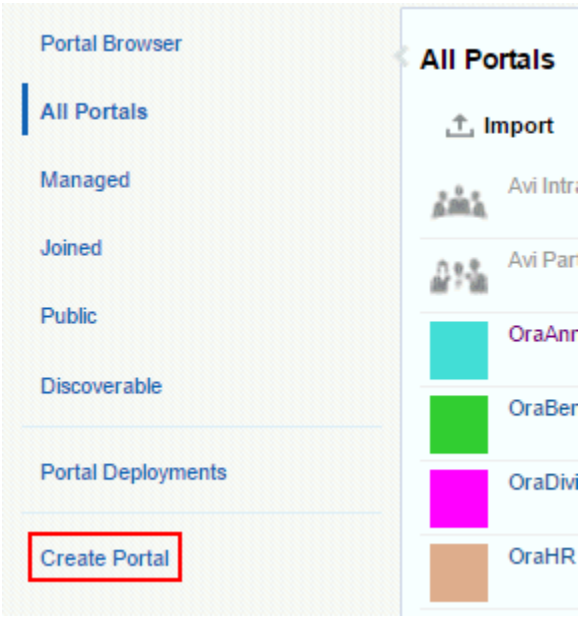
- **Last Updated** to order from most to least recently updated.
- **Member Count** to order by greatest to least number of portal members.
- **Portal Name** to order alphabetically by internal name of the portal, as specified by its **Name** value in the portal administration. The internal name is not visible on the **Portals** page.

Creating a Portal

To create a new portal:

- On the **Portals** administration page, click **Create Portal** in the left pane.

Figure 36-5 Creating a New Portal



The **Select a Portal Template** page appears.

For information about creating a portal, see Creating and Building a New Portal in *Building Portals with Oracle WebCenter Portal*.

Exporting and Importing a Portal

With `Portals: Manage Security and Configuration` permission, you can export and importportals. For more information, see:

- Exporting Online Portals to an Archive Using WebCenter Portal Administration
- Importing a Portal from an Archive Using WebCenter Portal Administration

See also Troubleshooting Individual Portal and Portal Template Import and Export.

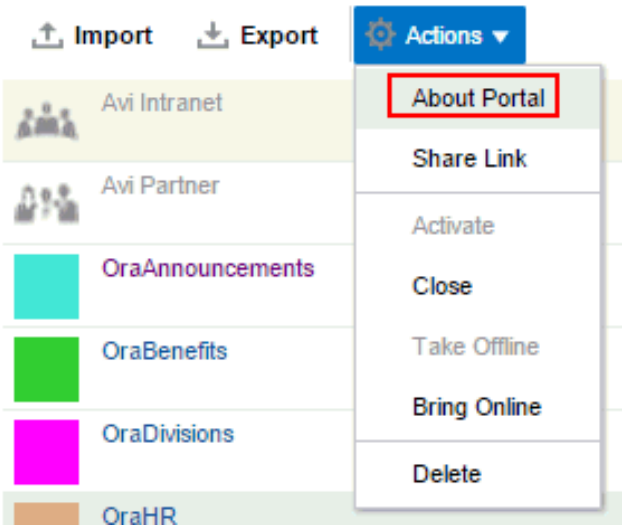
Viewing Information About Any Portal

On the **Portals** administration page, 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:

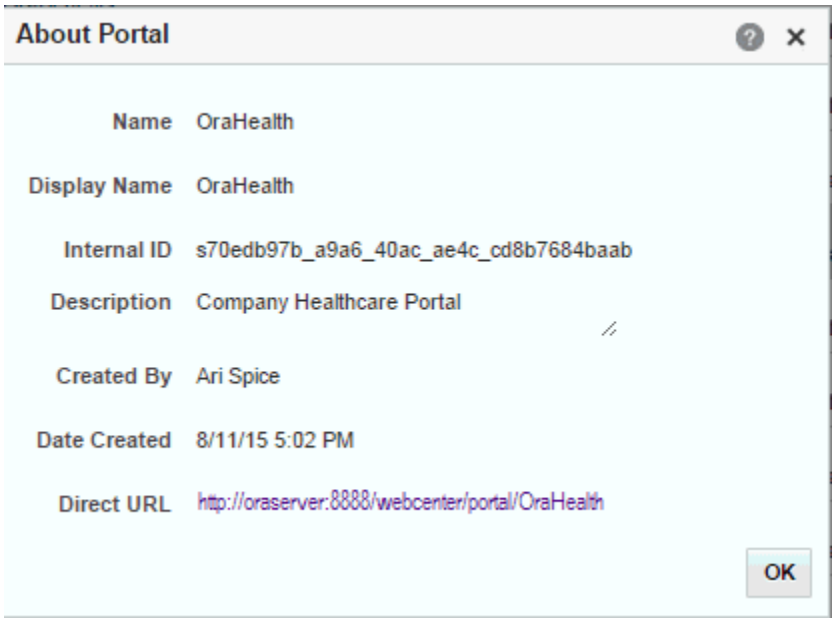
1. On the **Portals** administration page, select a portal by highlighting its row in the table.
2. From the **Actions** menu, select **About Portal**.

Figure 36-6 Viewing Information About a Portal



3. Explore the information in the About Portal dialog:

Figure 36-7 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.

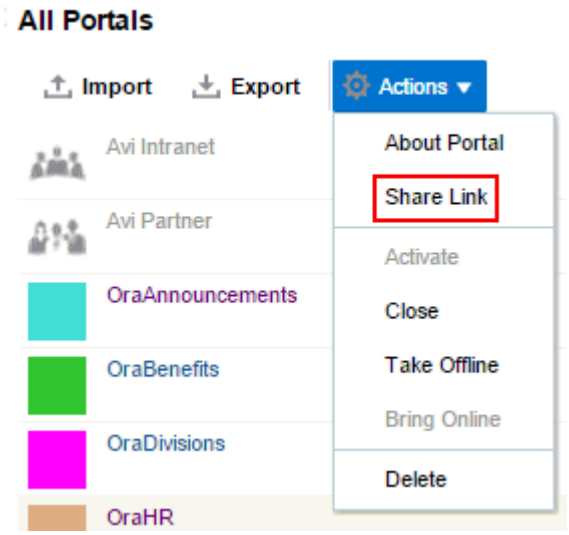
Sharing the Link to a Portal

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:

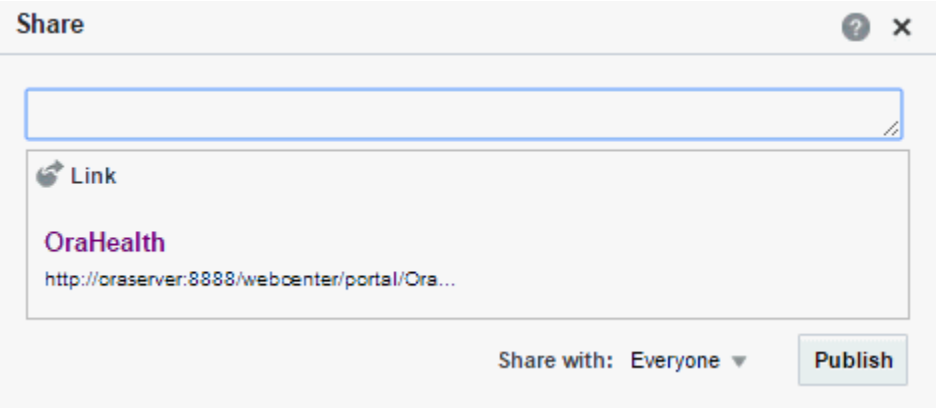
1. On the **Portals** administration page, select the required portal by highlighting its row in the table.
2. From the **Actions** menu, select **Share Link**.

Figure 36-8 Sharing a Link to a Portal



3. In the Share dialog, optionally enter a comment to appear with the link.

Figure 36-9 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**.

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.

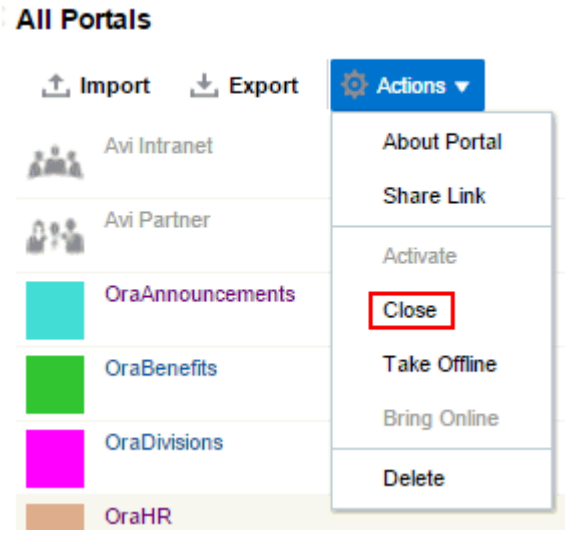
If you want to close down a portal temporarily, take the portal offline instead.

To close a portal:

1. On the **Portals** administration page, 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**.

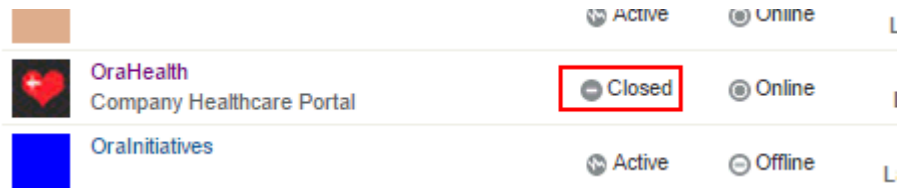
Figure 36-10 Closing a Portal



3. Confirm the action by clicking **OK**.

   Notice that the **Active** status changes to **Closed**.

Figure 36-11 Closed Portal Status



Reactivating Any Portal

You may close a portal if it is no longer being used. If you want to reopen a portal, you can reactivate it.

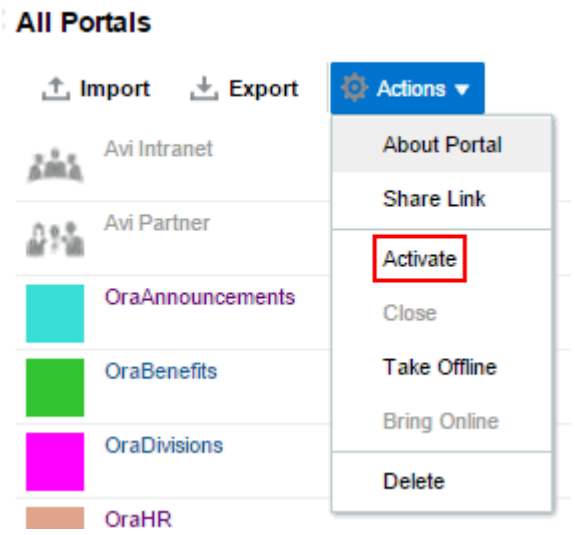
To reactivate a portal:

1. On the **Portals** administration page, 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 36-12 Activating a Portal



3. Confirm the action by clicking **OK**.

   Notice that the **Closed** status changes to **Active**.

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. With `Portals: Manage Security and Configuration` permission, you can access a portal that is offline, or bring it back online. Without this permission, users see the Portal Unavailable page (see Customizing System Pages).

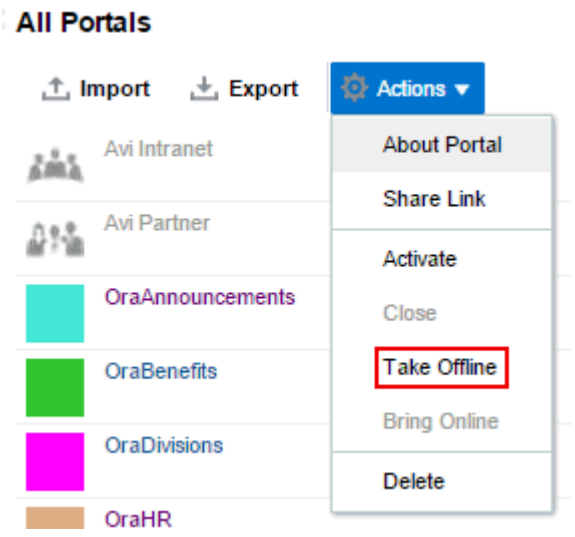
To take a portal offline:

1. On the **Portals** administration page, 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**.

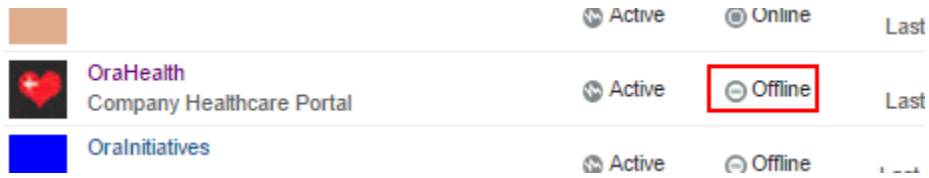
Figure 36-13 Taking a Portal Offline



3. Confirm the action by clicking **OK**.

   Notice that the **Online** status changes to **Offline**.

Figure 36-14 Offline Portal Status



Bringing Any Portal Back Online

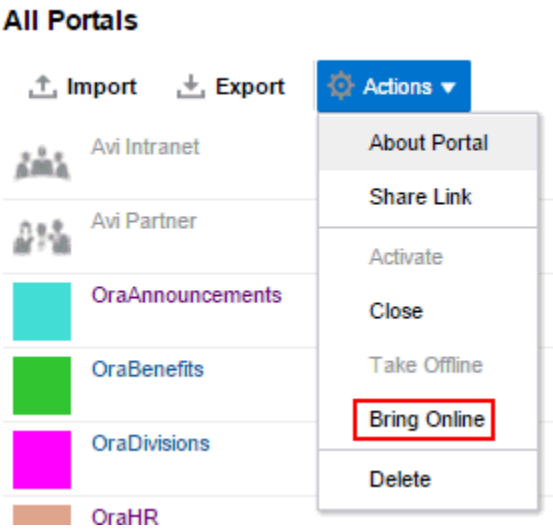
To bring any portal back online:

1. On the **Portals** administration page, 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 36-15 Bringing a Portal Online



3. Confirm the action by clicking **OK**.

   Notice that the **Offline** status changes back to **Online**.

Deleting a Portal

When a portal has been closed or inactive for some time, you may want to remove it permanently from WebCenter 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.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

- The portal mail distribution list that is automatically created by the Oracle WebCenter Portal is deleted. However, distribution lists that are customized by the portal manager are not deleted.

- Content managed by external services, such as content repositories and mail, is removed.

You cannot delete a portal while the portal manager is editing portal settings, but there are no other restrictions.

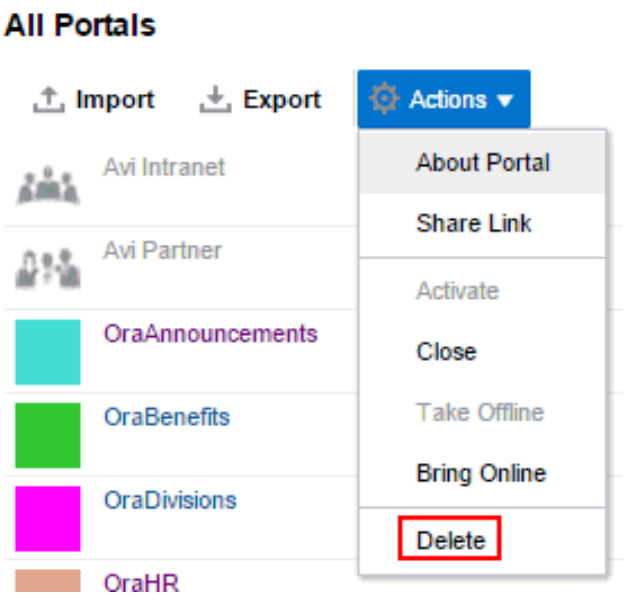
To delete a portal:

1. On the **Portals** administration page, 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 36-16 Deleting a Portal



3. Click **Delete** to confirm that you want to delete the portal(s).

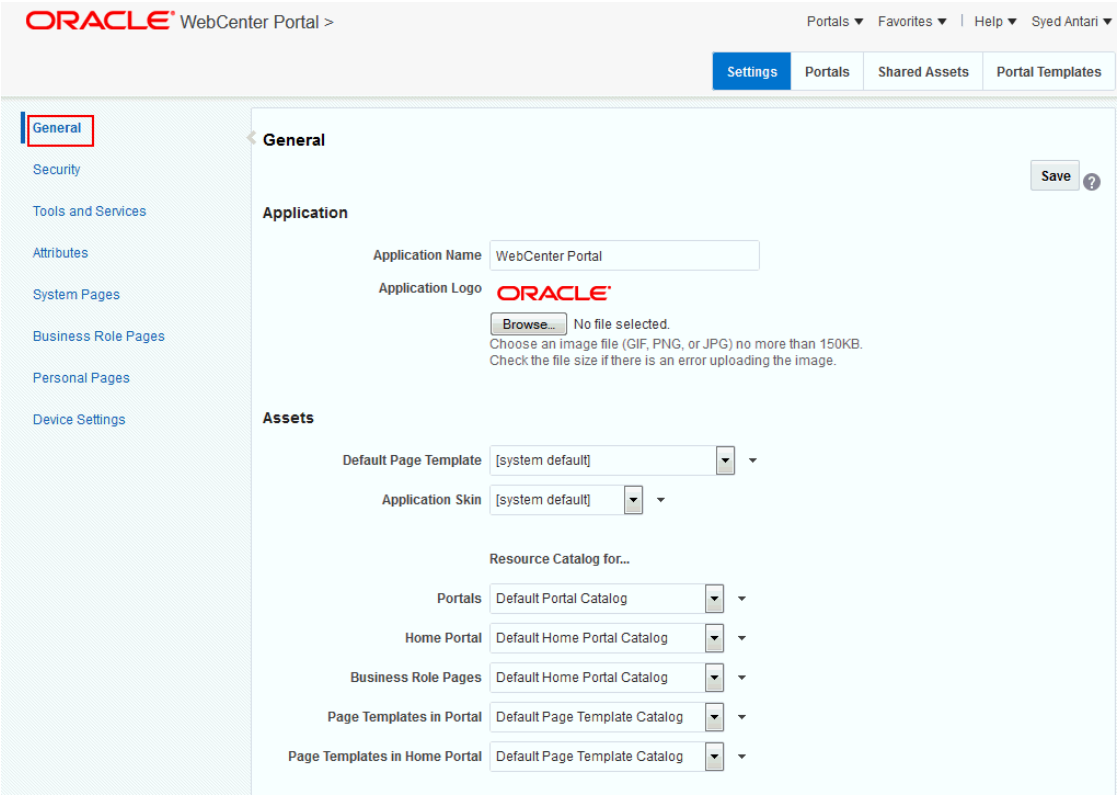
If the delete process fails for any reason, the portal is not removed from the **Portals** page; this sometimes happens when a back-end server cannot be contacted. If you click **Delete** again, the portal is removed.

37
Configuring Global Defaults Across Portals

Use the **General** page in WebCenter PortalAdministration to modify default settings such as the default page template and skin across all portals to suit the needs of the organization.

> **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 All`
> - `Portal Server: Manage Configuration`

Figure 37-1 WebCenter Portal Administration: General Page



Topics:

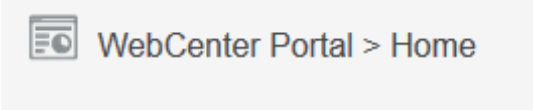
- Customizing the Name and Logo in the Home Portal

- Choosing a Default Page Template
- Choosing a Default Skin
- Choosing Default Resource Catalogs
- Customizing Copyright and Privacy Statements
- Customizing the Online Help Link
- Choosing a Default Display Language
- Redirecting on Logout
- Choosing a Default Start (or Landing) Page
- Specifying Session Timeout Settings
- Enabling Self-Registration
- Choosing a Default Look and Feel for New Pages
- Enabling and Disabling Access to the Home Portal
- Setting Up Defaults for WebCenter Portal Tools and Services

Customizing the Name and Logo in the Home Portal

Out-of-the-box, the Oracle logo and application name **WebCenter Portal** appear in the banner of the Home portal pages. You can change both the logo and name on the Home portal pages 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.

Figure 37-2 WebCenter Portal Name and Logo on the Home Portal



> **Note:**
>
> The changes you make to the Application Name and Application Logo on the **General** page in WebCenter Portal Administration will only affect pages in the Home portal. They will not affect the other WebCenter Portal Administration pages.
>
> To change the logo on portal pages, see Changing the Portal Icon and Changing the Portal Logo in *Building Portals with Oracle WebCenter Portal*.

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 Editing a Page Template in *Building Portals with Oracle WebCenter Portal*.

To change the name or logo for the Home portal:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **General**.

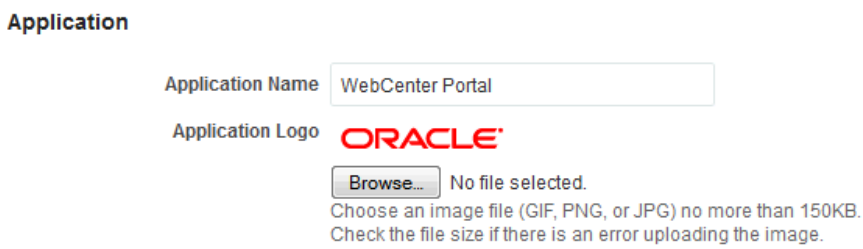
You can also enter the following URL in your browser to navigate directly to the **General** page:

`http://host:port/webcenter/portal/admin/settings/general`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. In the **Application Name** field, enter the new name.

Figure 37-3 Customizing the Application Name and Logo



3. To change the logo, click **Choose File** next to the **Application Logo** field.
4. In the Open 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 logo is uploaded to WebCenter Portal's image directory (`/webcenter/images`) .
5. Click **Save**.

   To confirm your changes, navigate to the Home portal to see the new logo in the top left corner of the banner area.

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 managers 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 be overridden from **Device Settings** in WebCenter Portal Administration. Edit the appropriate device group, as described in Editing a

Device . 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 Working with Skins in *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 Setting a Page Template's Preferred Skin in *Building Portals with Oracle WebCenter Portal*.

See also Working with Page Templates in *Building Portals with Oracle WebCenter Portal*.

To select the default page template for WebCenter Portal:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **General**.

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

   `http://host:port/webcenter/portal/admin/settings/general`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

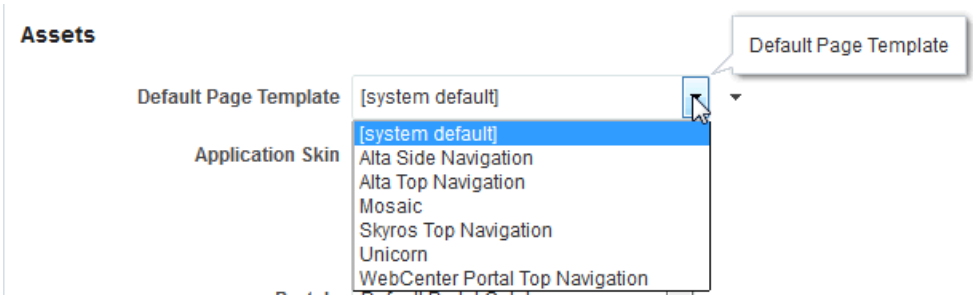
2. Do one of the following:

   - Select a **Default Page Template** from the available list.

> **Note:**
>
> `[system default]` specifies the default page template defined for WebCenter Portal, hardcoded in *webcenter-config. xml*

To learn how to add page templates to this list, see Publishing or Hiding a Page Template in *Building Portals with Oracle WebCenter Portal*.

Figure 37-4 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 Expression Language Expressions in *Developing for Oracle WebCenter Portal*.

  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**.

Choosing a Default Skin

As a system administrator, you can 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 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 Changing the Look and Feel of Your View in *Using Portals in 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 Creating a Skin in *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 Expression Language Expressions in *Developing for Oracle WebCenter Portal*.

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 Setting the Default Skin for a Portal in *Building Portals with Oracle WebCenter Portal*.

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 **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **General**

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

   `http://host:port/webcenter/portal/admin/settings/general`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Do one of the following:

   - Select an **Application Skin** from the available list.

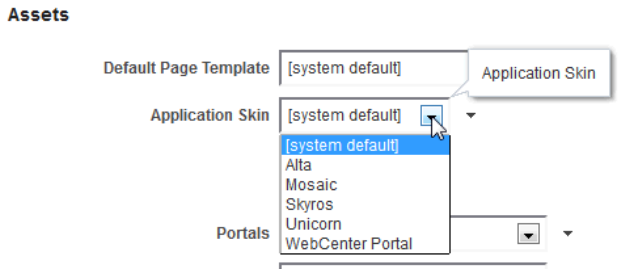
     > **Note:**
     >
     > If the desired skin does not appear in the **Application Skin** list, its Available option may be deselected. See Managing a Skin in *Building Portals with Oracle WebCenter Portal*.

     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 Setting a Page Template's Preferred Skin in *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 37-5 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.

Choosing Default Resource Catalogs

In WebCenter Portal, a resource catalog displays when you edit a page, page template, page style, or task flow asset and click **Add Content**. A resource 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 current user. Available resources include task flows, portlets, and page components, such as images, text, and hyperlinks.WebCenter Portal provides several built-in default resource catalogs, but you can add new task flows, remove task flows, or reorganize the folder hierarchy to better suit your audience or create new custom resource catalog from scratch. For details, see Working with Resource Catalogs in *Building Portals with Oracle WebCenter Portal*.

System administrators can specify the *default resource catalog* to be used for pages, page templates, page styles, and task flow assets in:

- New portals

- Home portal
- Business role pages
- Page templates in portals
- Page templates in the Home portal

To select default resource catalogs:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **General**.

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

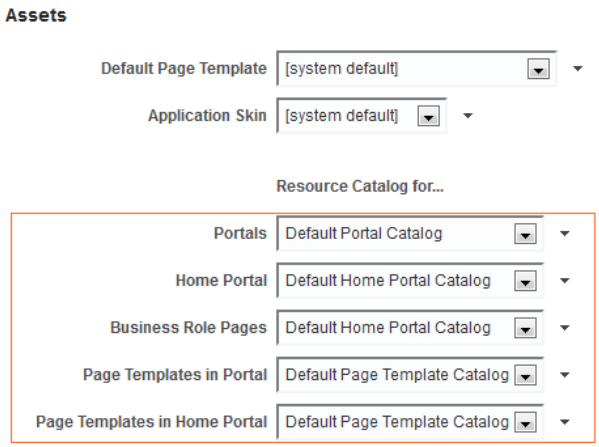
   `http://host:port/webcenter/portal/admin/settings/general`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Select default resource catalogs in the lists below **Resource Catalogs for...**.

   To learn how to expose existing resource catalogs in these lists, see Showing and Hiding Portal Assets in *Building Portals with Oracle WebCenter Portal*.

Figure 37-6 Selecting Resource Catalogs



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 Expression Language Expressions in *Developing for Oracle WebCenter Portal*.
4. Click **Save**.

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 37-7).

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 Expression Language Expressions in *Developing for Oracle WebCenter Portal*.

Figure 37-7 Copyright and Link to Privacy Statement in Page Footer

Individual portals may provide their own copyright and privacy statements. For details, see Customizing the Copyright Statement and Privacy URLin *Building Portals with Oracle WebCenter Portal*.

To customize or hide copyright and privacy statements:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **General**.

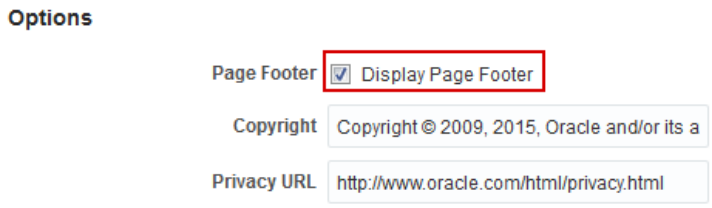
   You can also enter the following URL in your browser to navigate directly to the **General** page:

   `http://host:port/webcenter/portal/admin/settings/general`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Select or deselect **Display Page Footer** to display or hide copyright and privacy information in the page footer.

Figure 37-8 Customizing the Copyright and Privacy URL



Modify the legal notice and privacy URL as appropriate:

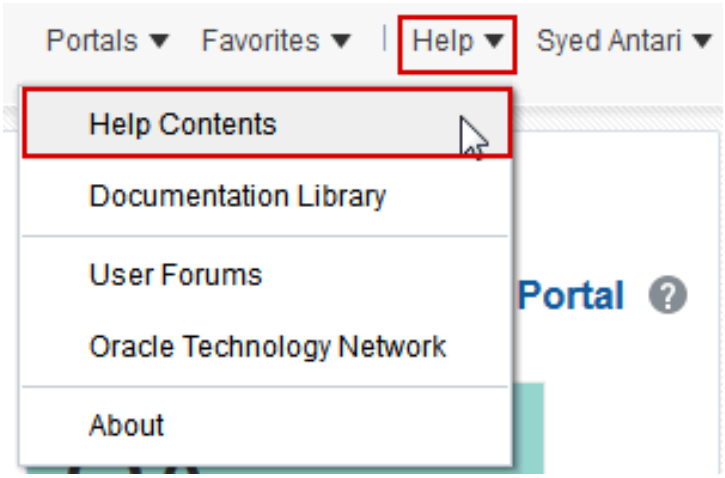
- **Copyright** - Enter a suitable copyright statement for your application. If no copyright information is required, leave this field blank.
- **Privacy URL** - Specify the location of the application's privacy policy. Enter a fully qualified URL. If no privacy information is required, leave this field blank.

3. Click **Save**.

Customizing the Online Help Link

System administrators can specify a URL to custom online help to replace the default WebCenter Portal online help that is accessed from the **Help** menu.

Figure 37-9 Help Link for WebCenter Portal



Out-of-the-box, the **Help Contents** link opens Oracle's built-in help. See Global Help in *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 "Expression Language Expressions in *Developing for Oracle WebCenter Portal*.

When you customize the **Help Contents** link, built-in help for WebCenter Portal is still available through help buttons and icons.

To customize the **Help Contents** link:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **General**.

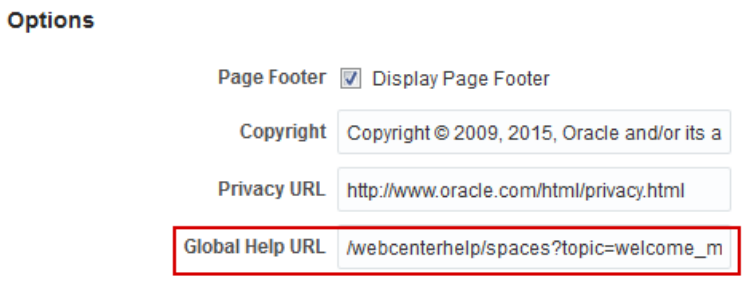
   You can also enter the following URL in your browser to navigate directly to the **General** page:

   `http://host:port/webcenter/portal/admin/settings/general`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. In the **Global Help URL** field, enter the URL to the location of your custom online help.

Figure 37-10 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 for WebCenter Portal is `http://www.oracle.com/pls/topic/lookup?ctx=fmw122130&id=GUID-AECE2F61-2727-4DB0-B74B-4BF279570BDD`. Enter this URL if you want to return to the default setting.

> **Note:**
>
> If you leave the **Global Help URL** field blank, the **Help Contents** link does not appear.

3. Click **Save**.
4. From the WebCenter Portal **Help** menu, select **Help Contents** to confirm that your custom help opens correctly.

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.

The display languages that are available for selection are also offered to users and portal managers through user preferences. As the system administrator, you can reduce the range of languages available to users. For details, see Customizing the Language List.

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 Expression Language Expressions in *Developing for Oracle WebCenter Portal*.

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 Setting a Portal Display Language in *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 Display Language Precedence in *Building Portals with Oracle WebCenter Portal*.

> **Note:**
>
> Portal Managers 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 Setting a Portal Display Language in *Building Portals with Oracle WebCenter Portal*.

To select the default display language for WebCenter Portal:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **General**.

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

   `http://host:port/webcenter/portal/admin/settings/general`

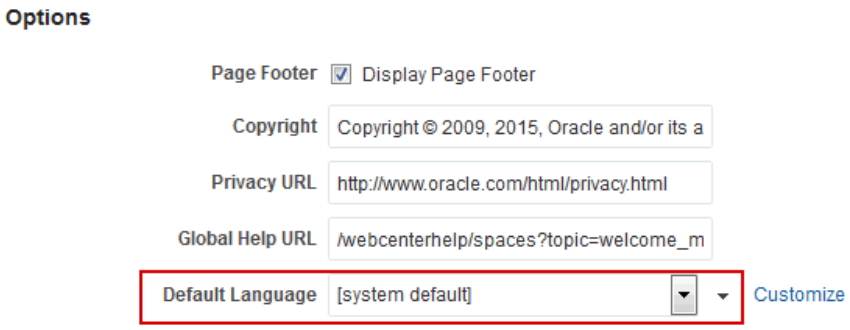
   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Do one of the following:

- Select a **Default Language** from the list.

  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 Customizing the Language List.

  Figure 37-11 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 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 Expression Language Expressions in *Developing for Oracle WebCenter Portal*.

3. Click **Save**.

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 **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **General**.

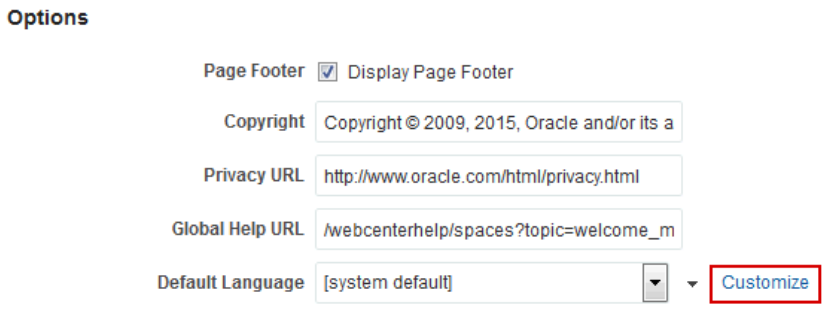
   You can also enter the following URL in your browser to navigate directly to the **General** page:

   `http://host:port/webcenter/portal/admin/settings/general`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

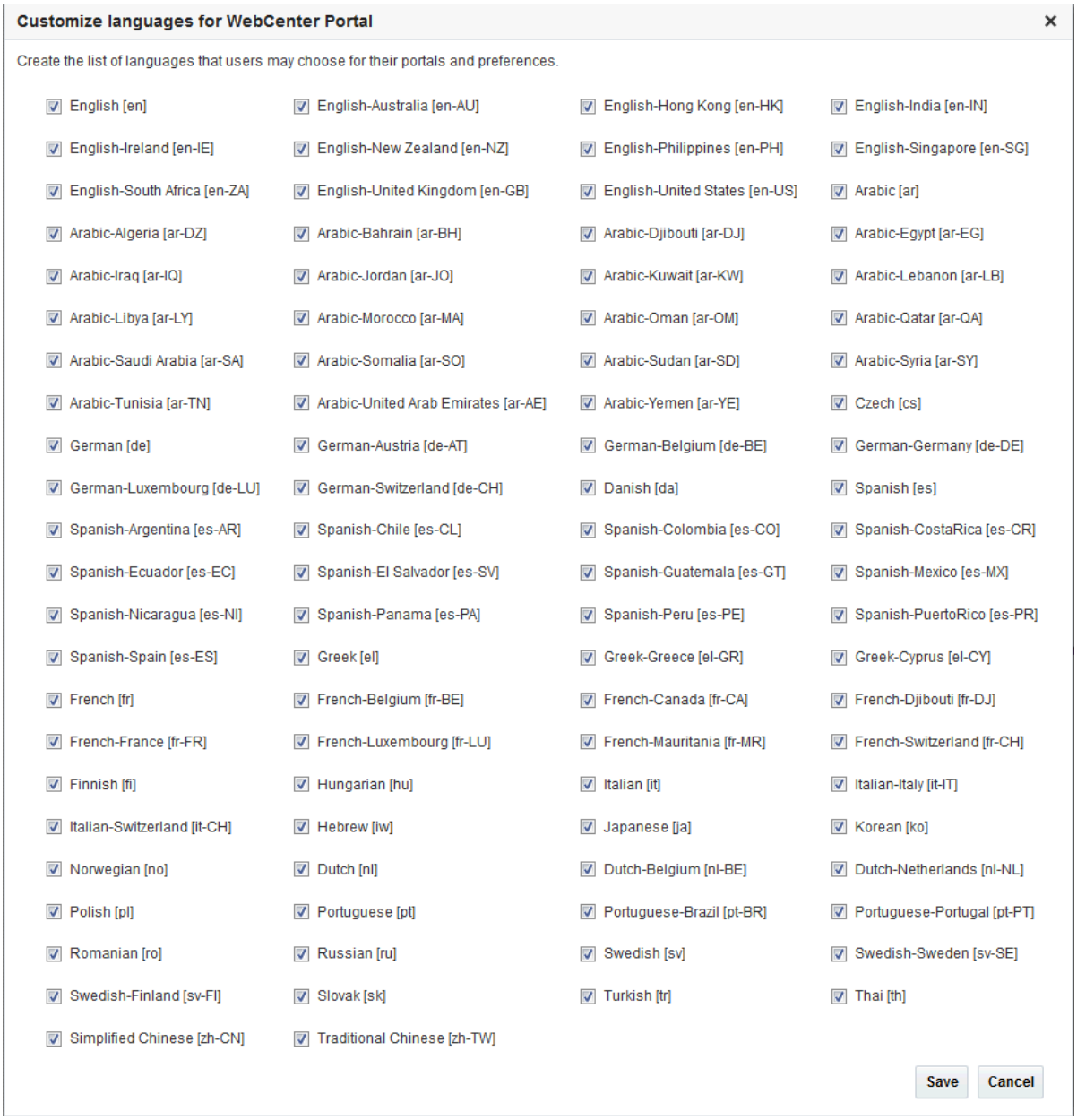
2. Click **Customize** next to **Default Language** (Figure 37-12).

Figure 37-12 Customize Option for Default Language



3. Select which languages to offer by selecting (or deselecting) each language check box (Figure 37-13).

Figure 37-13 Selecting Which Languages Are Available



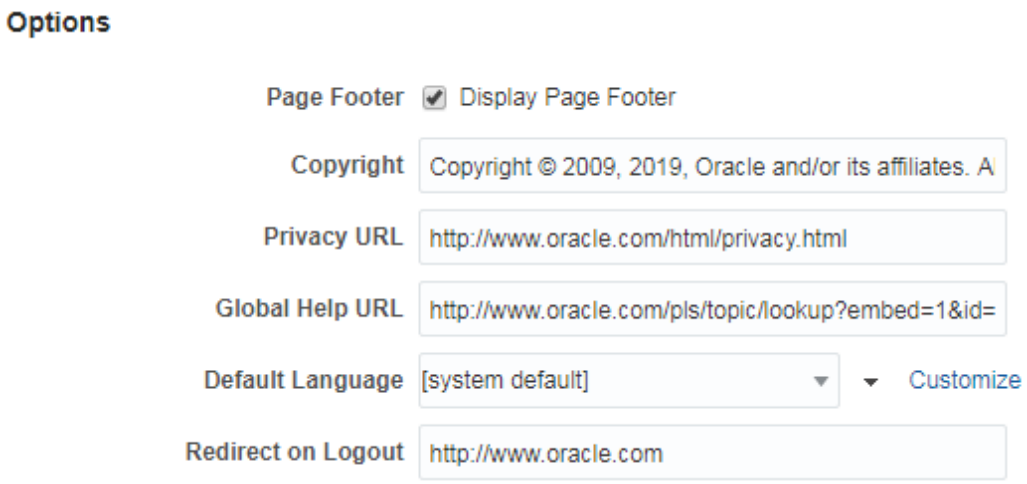
4. Click **Save**.

Redirecting on Logout

When you log off from the WebCenter Portal instance, you will be redirected to the URL that you have mentioned in the field **Redirect on Logout**. You need to specify only the white listed URLs that are mentioned in the `valid-url-link.xml` metadata file (placed at `/oracle/webcenter/webcenterapp/metadata`). For details, see Adding a List of Valid External URLs.

As shown in the figure, on logging out, you will be redirected to an Oracle site, *http://www.oracle.com*.

Figure 37-14 Redirecting on logging off



Choosing a Default Start (or Landing) Page

By default, users see the portal browser when they log in, but you can change the initial landing page to be the Home portal, a specific portal, or a specific page. You can specify a start page for a specific group, for authenticated users, and for public users.

The system administrator can configure the landing page for WebCenter Portal as shown in Table 37-1:

Table 37-1 WebCenter Portal Landing Page Behavior

Landing Page URL	Authenticated Users or Groups	Public Users
`/webcenter` or `/webcenter/portal`	Directed to the configured authenticated landing page	Directed to an unauthenticated landing page
`/webcenter/portal/`*`portal_name`*`/`*`page_name`*	• If the resource is public or user has access to the resource, directed to the resource page • If resource does not exist or user does not have access, then `Page Not Found` error is displayed	If resource is not public or does not exist, directed to the login page

To specify the landing page for WebCenter Portal:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **General**.

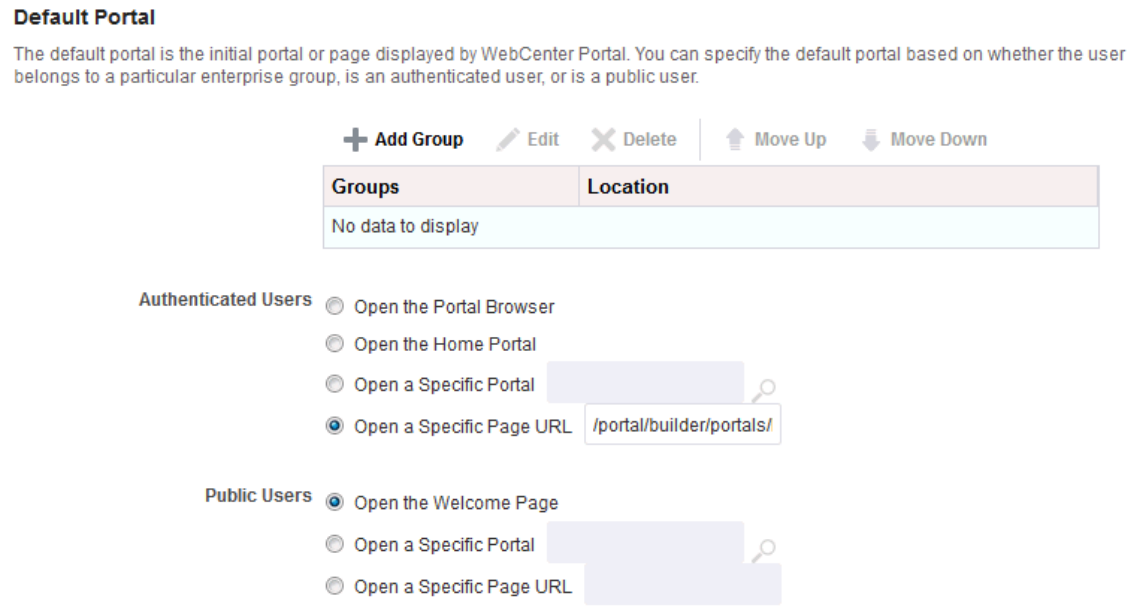
   You can also enter the following URL in your browser to navigate directly to the **General** page:

   `http://`*`host`*`:`*`port`*`/webcenter/portal/admin/settings/general`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Scroll to the **Default Portal** section.

Figure 37-15 Choosing a Default Start Page



3. Select what users see first when they log in:
   - To specify a default landing page for selected groups, see Specifying 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 Specifying a Default Start Page for Authenticated Users.
   - To specify a default landing page for all public users, see Specifying a Default Start Page for Public Users.
4. Click **Save**.

Specifying a Default Start Page for Groups

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **General**.

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

   `http://host:port/webcenter/portal/admin/settings/general`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

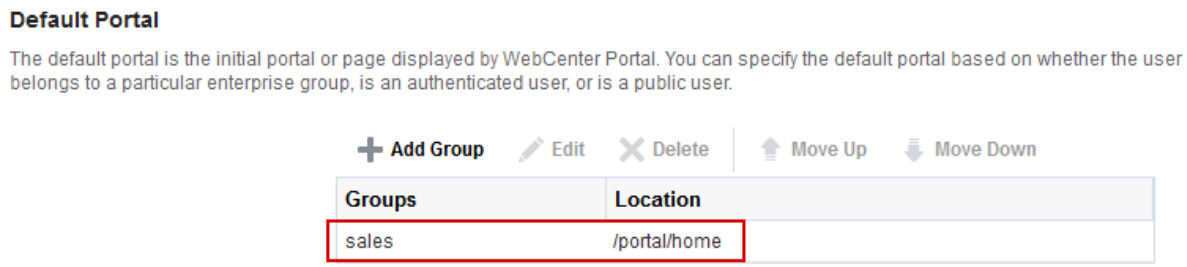
2. Scroll to the **Default Portal** section.
3. Click **Add Group** if you want selected enterprise groups to see a specific start page.

> **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.

4. From the Add Group dialog, search for a group or select a group from the list, then click **OK**.

   The selected group is added to the table.

Figure 37-16 Specifying a Landing Page for a Group

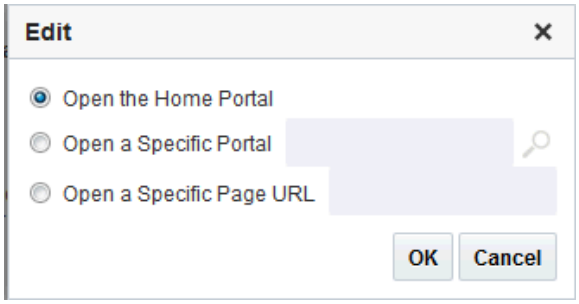


Any user belonging to the group will be directed to the default landing page upon logging in to WebCenter Portal. Note that by default, the landing page is set to the portal browser.

5. To change the **Location** of the landing page, select the group name and click **Edit**.

   The Edit dialog opens.

Figure 37-17 Landing Page Options



6. Select whether the group will first see the Home portal, or a specific portal or page:

> **Note:**
>
> Make sure that the specified page or portal is available to all users (see Setting Page Security in *Building Portals with Oracle WebCenter Portal*). If a user does not have access to the specified page or portal, the `Page Not Found`

- **Open the Home Portal**. Select to specify that 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`

  `/portals/portalname/page/landingpage`

  If you specify an external page, make sure that you specify the full URL.

7. Click **Save**.

Specifying a Default Start Page for Authenticated Users

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **General**.

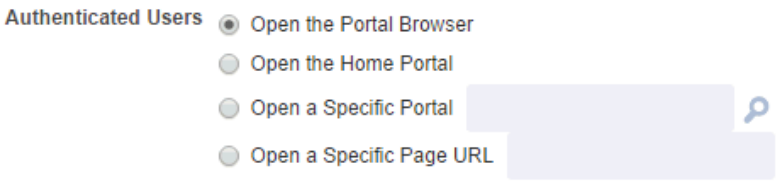
   You can also enter the following URL in your browser to navigate directly to the **General** page:

   `http://host:port/webcenter/portal/admin/settings/general`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Scroll to the **Default Portal** section.
3. In the **Authenticated Users** section, specify what authenticated users who are not in any of the specified groups see when they first log in.

Figure 37-18 Selecting a Landing Page for Authenticated Users

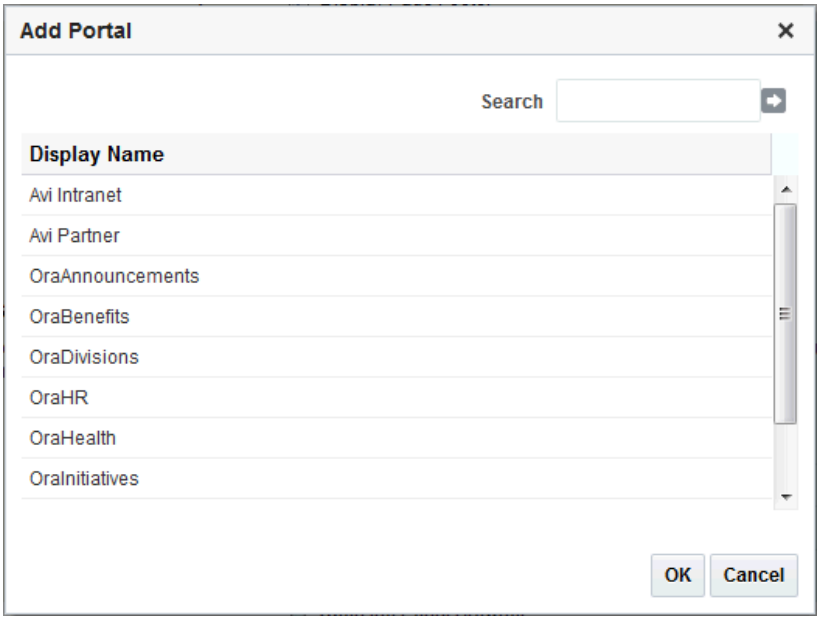


- **Open the Portal Browser**. Selected by default. Users see the portal browser when they first log in.
- **Open the Home Portal**. Select to specify that users see the Home portal when they first log in.
- **Open a Specific Portal**. Select to specify that a particular portal displays, and enter the portal name For example:

  `http://host:port/webcenter/portal/portalName`

  Or click **Browse** to select from a list of portals (Figure 37-19). Select an option from the **Add Portal** list, and click **OK**, or enter the portal name in the **Search** field and click **Search**.

Figure 37-19 Add Portal Dialog



- **Open a Specific Page URL**. Select to specify that a particular page displays, and enter the page location.

> **Note:**
>
> Make sure that the specified page is available to all users (see Setting Page Security in *Building Portals with Oracle WebCenter Portal*). If a user does not have access to the specified page, the `Page Not Found` message displays.

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.

4. Click **Save**.

Specifying 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 Managing Roles and Permissions for a Portal in *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 Granting Public Access to a Portal in *Building Portals with Oracle WebCenter Portal*.

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **General**.

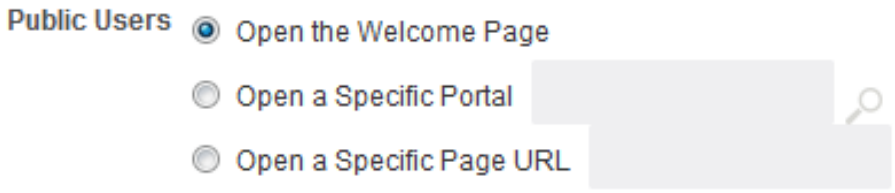
   You can also enter the following URL in your browser to navigate directly to the **General** page:

   ```
   http://host:port/webcenter/portal/admin/settings/general
   ```

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. In the **Default Login Settings** section, specify what public users see when they first log in.

Figure 37-20 Selecting a Landing Page for Public Users



- **Open the Welcome Page**. Selected by default. Users see the WebCenter Portal welcome page 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. 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**.

Specifying Session Timeout Settings

When there is no activity for an extended period of time in a WebCenter Portal session, it times out. 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.

Out-of-the box, the WebCenter Portal session timeout is set to 20 minutes. When deploying shared libraries, additional paths might be added to the `/webcenter` root context (for example, `/webcenter/images/`). In such instances where URLs based on the `/webcenter` root context are not handled by WebCenter Portal, the HTTP session timeout is set to 24 hours.

> **Note:**
>
> An application-level new custom attribute is added in WebCenter Portal to set the Windows timeout. The value of the attribute `wcAdfLastWindowSessionTimeout` is currently set to 10 minutes. See Editing a Global Attribute. This is to avoid getting an error page when a page is refreshed after some time of inactivity.

To modify the session timeout settings for WebCenter Portal:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **General**.

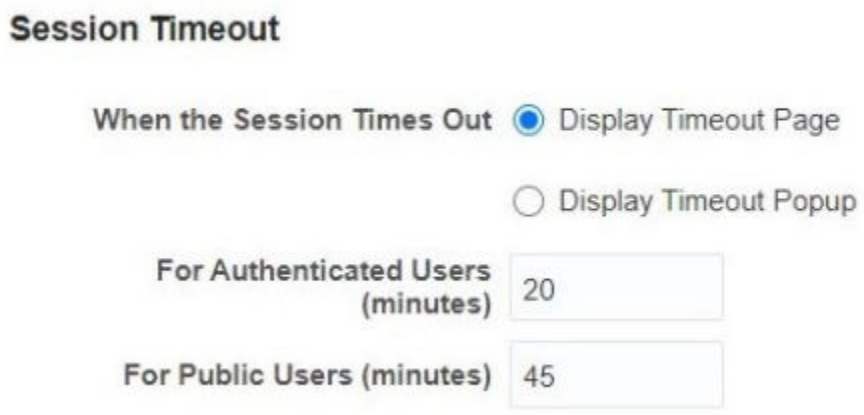
   You can also enter the following URL in your browser to navigate directly to the **General** page:

   `http://host:port/webcenter/portal/admin/settings/general`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

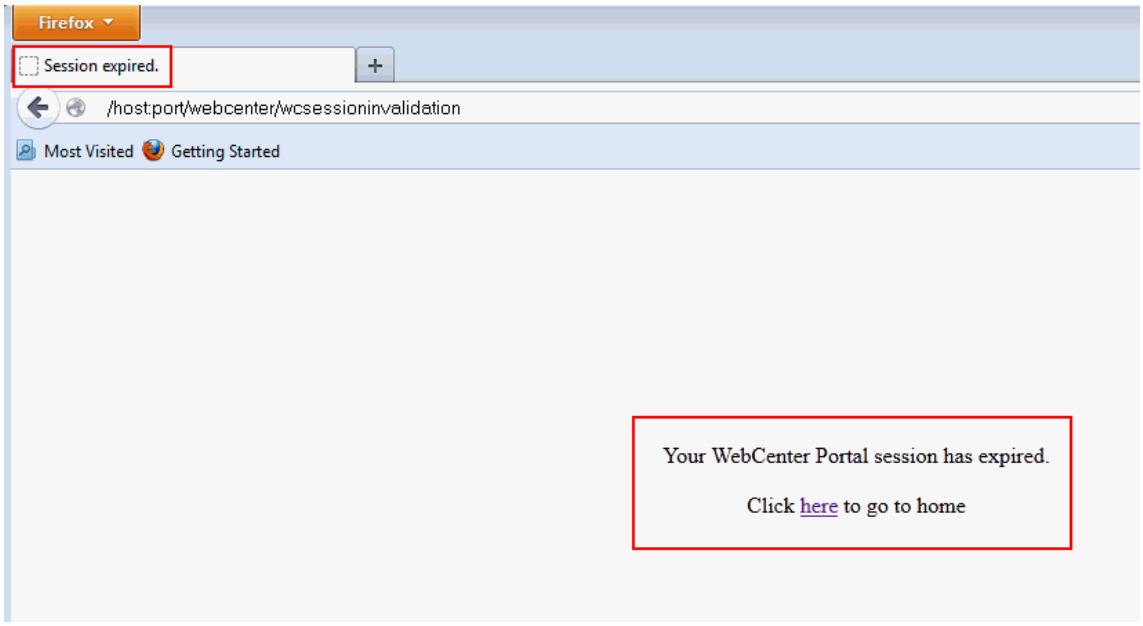
2. Scroll to the **Session Timeout** section.
3. Select the desired result when WebCenter Portal times out:

Figure 37-21 Session Timeout Options



- **Display Timeout Page**. Select to display the WebCenter Portal timeout page in the browser, where the user can click the provided link to log in again and restart at the default start page.

Figure 37-22 Timeout Page



- **Display Timeout Popup**. Select to display an Expiration Warning notification popup 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, 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 37-23 Expiration Warning Notification (displays at Session Timeout)

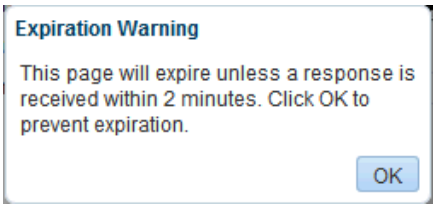
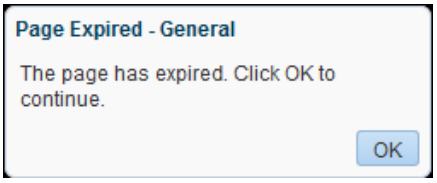


Figure 37-24 Timeout Notification (displays 2 minutes after Session Timeout)



4. In the **For Authenticated Users (minutes)** field, enter a new value.

The default value is 20 minutes, the minimum value is 5, and the maximum value is 1440 (24 hours). If this field is left blank, the default value (20) applies.

> **Note:**
>
> If 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 Portal session will be automatically created as long as the SSO timeout is not reached.

5. In the **For Public Users (minutes)** field, enter a new value. When the session timeout value for Public User is reached, the expiration warning notification popup is displayed. 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, the user can click **OK** to trigger the page refresh and extends the session by the value specified in minutes. After the refresh, the user continues to remain on the same page.
6. Click **Save**.

Enabling Self-Registration

A system administrator can enable WebCenter Portal self-registration. Through self-registration, 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:

- About Self-Registration
- Enabling Self-Registration By Invitation-Only
- Enabling Anyone to Self-Register

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 any public user is allowed to self-register, a **Register for an account** link displays on the WebCenter Portal sign in page. To enable this feature, see Enabling Anyone to Self-Register.

Self-registration by invitation is available too. This feature allows portal managers 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 portal manager must approve their subscription request before gaining access to the portal. If the portal is public or further approval is not

required, the new user gains access to the portal immediately. See Enabling Self-Registration By Invitation-Only.

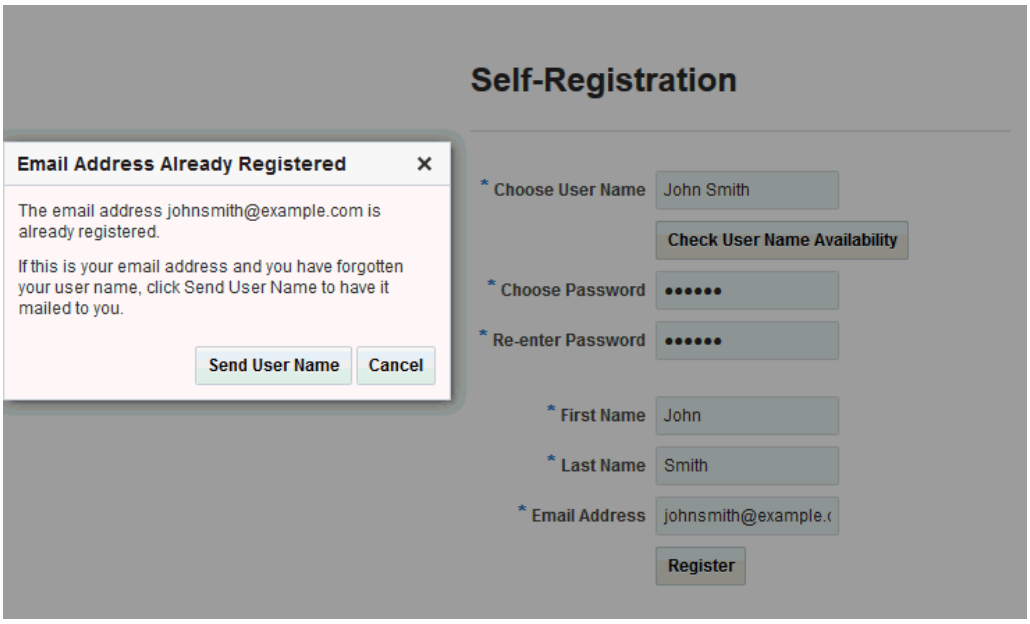
> **Note:**
>
> If self-registration is not enabled for WebCenter Portal, identity store management takes place through the WLS Administration Console (or directly into embedded LDAP identity stores using LDAP commands) and is the responsibility of the system administrator. See also Adding Users to the Embedded LDAP Identity Store.

A self-registration page is supplied with WebCenter Portal. Users with the `Administrator` role can add new components to the page and change the page layout if required. See Customizing System Pages.

> **Note:**
>
> While you can access the Self-Registration page using a pretty URL, the fields on the page are not active when accessed in this way. Fields are active only when non-registered users access the page by clicking the **Register** link on the WebCenter Portal Sign In page.

The self-registration page provided with WebCenter Portal offers to send a user name reminder mail message to anyone who tries to register with an email address that has already been used (Figure 37-25).

Figure 37-25 Email Address Already Registered



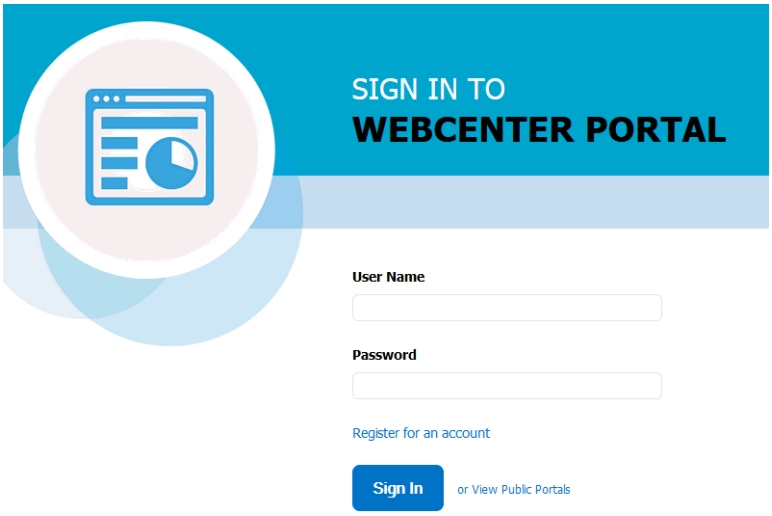
This feature only works if *public credentials* are defined for the external application that is providing authentication for the mail service in WebCenter Portal. If users experience issues with this feature, check the mail server connection and its associated external application connection are configured correctly and that public credentials are defined. See also Registering Mail Servers.

- For more information about setting up public credentials using Enterprise Manager, see Table 18-5 in Managing External Applications.
- For more information about setting up public credentials using WLST, see `addExtAppCredential` in *Oracle WebLogic Scripting Tool*.

Enabling Anyone to Self-Register

When any public user is allowed to self-register, a **Register for an account** link displays on the WebCenter Portal sign in page.

Figure 37-26 Self-Registration Available on Sign In Page



Users who self-register are added directly to the WebCenter Portal identity store and assigned the `Authenticated-User` role. By default, users with `Authenticated-User` role have access to the 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 Modifying Application Role Permissions.

To allow anyone to self-register with WebCenter Portal:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **General**.

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

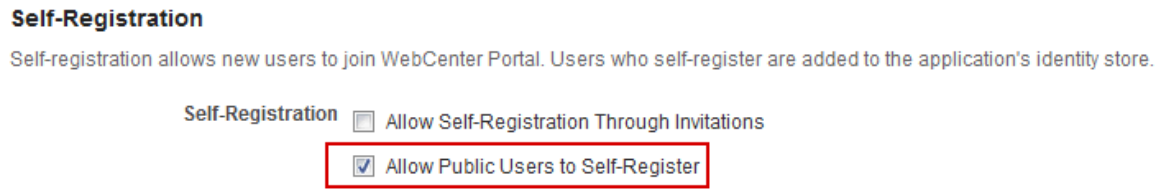
   `http://host:port/webcenter/portal/admin/settings/general`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Scroll to the **Self-Registration** section.
3. Select **Allow Public Users to Self-Register**.

   When you deselect this option, public users cannot self-register with WebCenter Portal. If you want to enable self-registration on an invitation-only basis, see Enabling Self-Registration By Invitation-Only.

Figure 37-27 Allowing Public Users to Self Register



4. Click **Save**.

Anyone with internet access can now register themselves as a user of the WebCenter Portal application, as described in Registering Yourself with WebCenter Portal in *Using Portals in Oracle WebCenter Portal*. If users experience no response when they attempt to register with WebCenter Portal, they should refresh their browser and try again.

Enabling Self-Registration By Invitation-Only

By default, 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.

A 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 option is enabled, portal managers can invite anyone to join their portal by sending them a customized 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`. By default, users with the `Authenticated-User` role have access to the 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 Modifying Application Role Permissions.

To allow anyone to self-register with WebCenter Portal through invitations:

1. Ensure that the prerequisite configuration is set up, as described in Configuring Self-Registration By Invitation in WebCenter Portal.
2. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **General**.

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

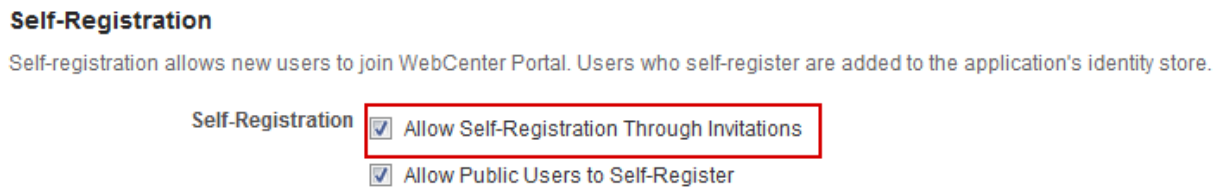
   `http://host:port/webcenter/portal/admin/settings/general`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

3. Scroll to the **Self-Registration** section.
4. Select **Allow Self-Registration Through Invitations**.

   When you deselect this option, only existing users are candidates for portal membership.

Figure 37-28 Allowing Self-Registration Through Invitations



5. Click **Save**.

After you enable this option, portal managers can invite anyone to become a member of their portal. See Inviting a Non-Registered User in *Building Portals with Oracle WebCenter Portal*.

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:

- Customizing System Pages
- Managing Business Role Pages
- Managing Personal Pages

Individuals may personalize the default settings in their Home portal view. For more information, see Setting Application-Level Page Creation Defaults for Personal Pages.

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 37-2 to determine which permission settings are required for the different roles.

To enable or disable access to the Home portal:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Security**.

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

   `http://host:port/webcenter/portal/admin/settings/security`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. On the **Roles** tab, select a role from the **Role** list, then under **Portal Server**, select or deselect the **View** check boxes for the roles (see Figure 37-29), as appropriate (see Table 37-2).

Figure 37-29 Portal Server: View Permissions

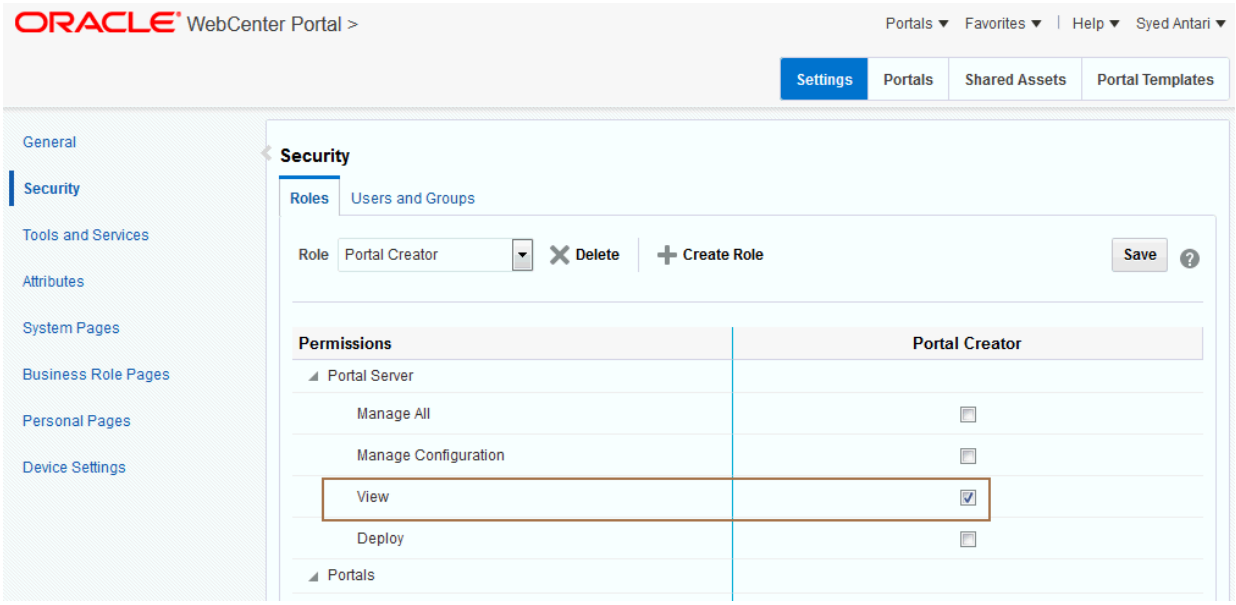


Table 37-2 Portal Server: View Permissions

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.)
`Portal Creator User`	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 **Save** to save your changes.

   New permissions are effectively immediately.

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 Managing Tools and Services.

Adding a List of Valid External URLs

Some WebCenter Portal components, such as links and wikis, allow portal developers and users to enter an external URL. This opens up the possibility to enter a malicious URL in the portal pages that are exposed to public users.

To provide security against this vulnerability, you must enter allowable external URLs in the `valid-link-url.xml` file. If an external URL is not listed in the file, the following error message appears:

The URL entered is not available in the list of valid URLs. Contact your system administrator for the list of valid URLs.

> **Note:**
>
> If the `valid-link-url.xml` file does not exist in `/oracle/webcenter/webcenterapp/metadata/`, you will need to create it.

To add a list of valid external URLs:

1. Export the `valid-link-url.xml` file from the MDS repository using the `exportMetadata` WLST command:

   ```
   exportMetadata(application='webcenter', server='WC_Portal',
   toLocation='MDS_FileStore_Location', docs='/oracle/webcenter/
   webcenterapp/metadata/valid-link-url.xml')
   ```

   > **Note:**
   >
   > If the following error appears, it indicates that the `valid-link-url.xml` file is not available in the metadata store.
   >
   > ```
   > MDS-01059: document with the name /oracle/webcenter/
   > webcenterapp/metadata/valid-link-url.xml missing in the
   > source metadata store.
   > 1 out of 1 documents failed to be transferred.
   > ```
   >
   > If the `valid-link-url.xml` file is missing, you need to create the file under the *`MDS_FileStore_Location`* directory with the following structure:
   >
   > `/oracle/webcenter/webcenterapp/metadata/valid-link-url.xml`

2. Add the list of allowed external URLs to the `valid-link-url.xml` file by adding an `Item` element as shown in the following example:

   ```
   <?xml version='1.0' encoding='UTF-8'?>
   <valid-link-url  xmlns="http://xmlns.oracle.com/webcenter/validurl">
   <Item link="http://www.oracle.com"/>
   <Item link="http://www.google.com"/>
   ```

```
<Item link="http://www.example.com"/>
</valid-link-url>
```

3. Save and import the `valid-link-url.xml` file to the MDS repository using the `importMetadata` WLST command:

```
importMetadata(application='webcenter',
server='WC_Portal',fromLocation='MDS_FileStore_Location',docs='/oracle/
webcenter/webcenterapp/metadata/valid-link-url.xml')
```

38
Managing Security Across Portals

Use the **Security** page in WebCenter Portal Administration to assign suitable permissions to user roles, assign users to roles, and create new custom roles as required.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

> **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 All`

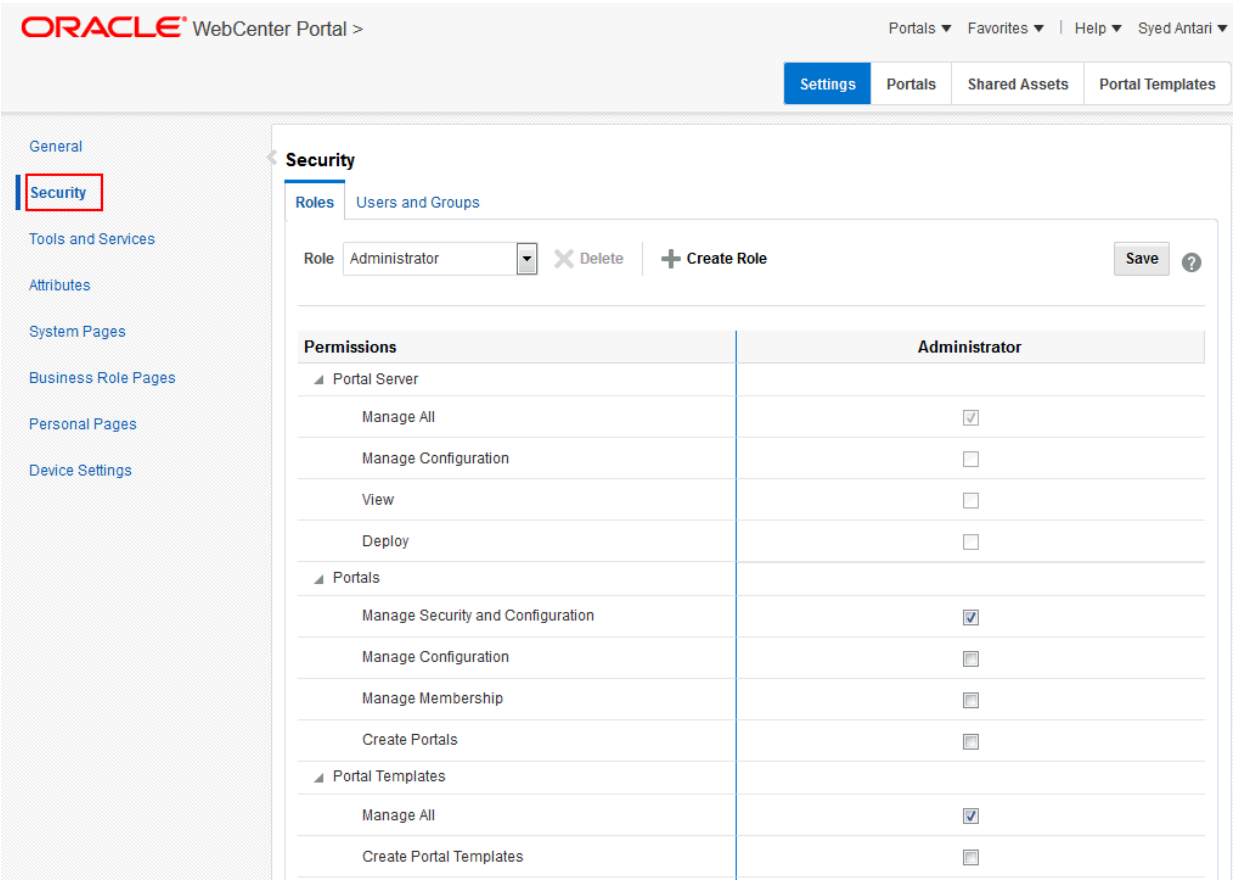
Topics:

- About WebCenter Portal Security
- About Users
- About Application Roles and Permissions
- About Roles and Permissions Within a Portal
- Managing Users
- Managing Application Roles and Permissions

About WebCenter Portal Security

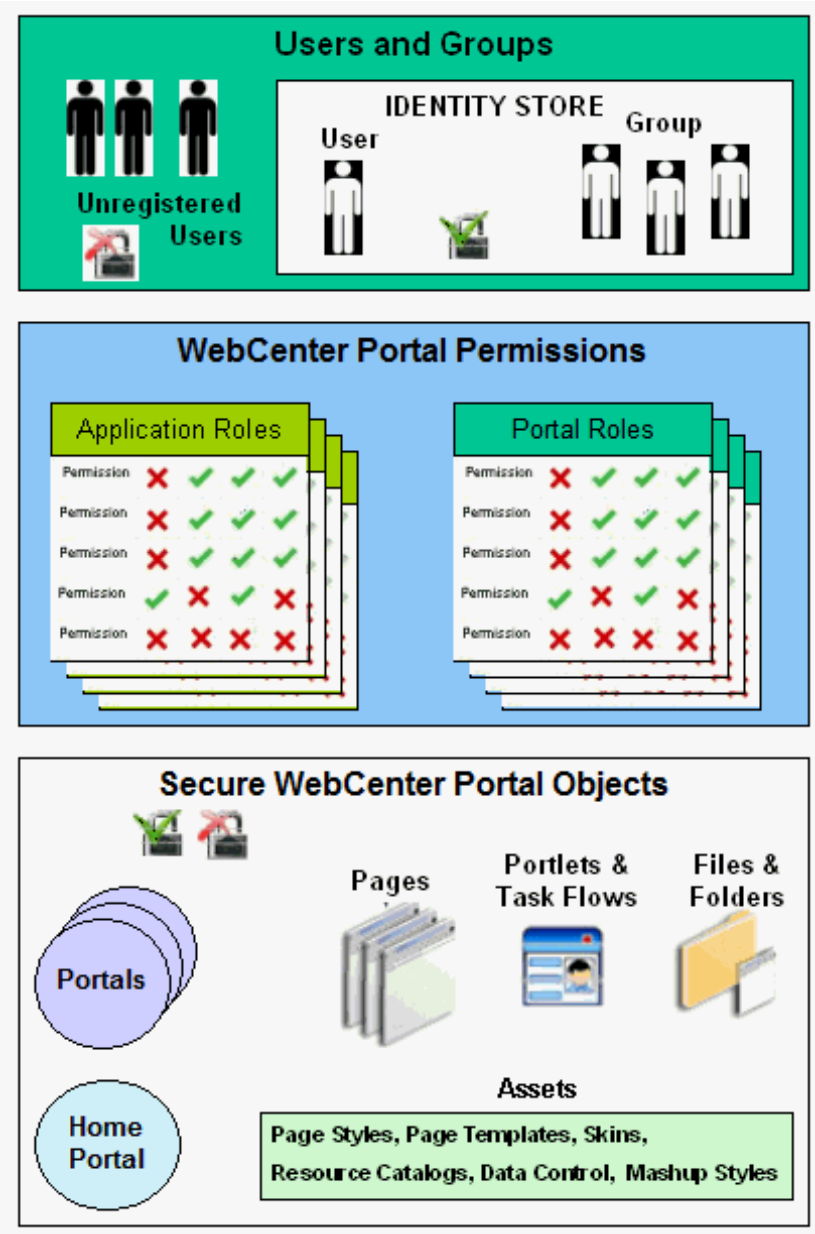
WebCenter Portal provides a comprehensive security model that enables you to control what users can see and change in WebCenter Portal. Using the **Security** page in WebCenter Portal Administration (Figure 38-1), you can control which users (and groups) have access to individual portals and the Home portal and you can also control exactly what users and groups can see and do by enabling and disabling various permissions.

Figure 38-1 WebCenter Portal 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 assets (such as page templates, page styles, skins, resource catalogs, and so on).

Figure 38-2 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

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.

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.

Assets

Various portal assets help define the overall structure, look and feel, and content in portals. These include page styles, page templates, skins, resource catalogs, Content Presenter display templates, task flow styles, data controls, and task flows. Users with appropriate privileges can build and customize assets for the entire application or individual portals.

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.

About Users

A WebCenter Portal user has a login account for WebCenter Portal—provisioned directly from an existing identity store. See 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 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 38-1 Default User 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.

About Application Roles and Permissions

Application roles control the level of access a user has to information and services in WebCenter Portal. Application role assignment is the responsibility of the system administrator. Administrators can assign users a default application role or create additional, custom roles specific to their application deployment. Every application role has specific, defined capabilities known as *permissions*. These permissions allow users to perform specific actions in the Home portal.

This section includes:

- About Application Roles
- About Application Permissions

About Application Roles

Application role assignment is the responsibility of the system administrator. Administrators can assign users a default application role or create additional, custom roles specific to their application deployment. For more details, see:

- Default Application Roles
- Custom Application Roles

Application roles apply when users are working in the Home portal or on application-level tasks. A different set of roles and permissions apply when a user is working within a particular portal. It is the portal manager's responsibility to determine suitable role assignments for each of its members. See also Managing Application Roles and Permissions, and Administering Security in a Portal in *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 Application Roles and Enterprise Roles.

Default Application Roles

WebCenter Portal provides several default application roles (Table 38-2). You cannot delete the default application roles of `Administrator`, `Public-User`, and `Authenticated-User`, but

you can modify the default permission assignments for each role. For more information, see Modifying Application Role Permissions.

Table 38-2 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 as well as deploy and propagate portal. Out-of-the-box, the system administrator is the only user assigned full administrative privileges for the WebCenter Portal through the `Administrator` role. **Note:** The `Administrator` role allows administration permissions on a private portal (such as managing membership), but does not allow access to a private portal's page contents.	Yes* *Except for Application permissions which are read-only
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. In order to manage membership of `AppConnectionManager` role, use the following options: • WLST commands: – `grantAppRole`: To add a user or role to the Connection Manager (For command syntax and examples, see grantAppRole in *Infrastructure Security WLST Command Reference*). – `revokeAppRole`: To remove the member from the Connection Manager (For command syntax and examples, see revokeAppRole in *Infrastructure Security WLST Command Reference*). • Enterprise Manager (see Managing Application Roles in *Securing Applications with Oracle Platform Security Services*. **Note**: You cannot view `AppConnectionManager` role in Oracle WebCenter Portal UI.	No

Table 38-2 (Cont.) Default Application Roles for WebCenter Portal

Application Role	Description	Modify?
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. In order to manage membership of `AppConnectionViewer` role, use the following option: • WLST commands: – `grantAppRole`: To add a user or role to the Connection Manager (For command syntax and examples, see grantAppRole in *Infrastructure Security WLST Command Reference*). – `revokeAppRole`: To remove the member from the Connection Manager (For command syntax and examples, see revokeAppRole in *Infrastructure Security WLST Command Reference*). • Enterprise Manager (see Managing Application Roles in *Securing Applications with Oracle Platform Security Services*. **Note**: You cannot view `AppConnectionViewer` role in Oracle WebCenter Portal UI.	No
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, resource catalogs, skins, task flow styles, and task flows; update People Connections data, and connect with people.	Yes
Portal Creator	Users with the `Portal Creator` role are assigned the `Portals: Create Portals` and `Portal Server: View` permission by default. Users in this role do not have the ability to manage or create portal templates. This role is provided to make sure that only a select few portal users have the ability to create portals. Upon creating a portal, the `Portal Creator` role inherits the permissions inherent in the portal-level `Portal Manager` role. Users in this role have the ability to import, export, and deploy portals (only if they are in a role that has the application level `Portal Server: Deploy` permission) that they are members of and those portals that they manage.	Yes

Table 38-2 (Cont.) Default Application Roles for WebCenter Portal

Application Role	Description	Modify?
Authenticated-User	Authenticated users of WebCenter Portal are granted the `Authenticated-User` role. Users who log in are assigned this role and, by default, have access to the Home portal, pages that they create, and public pages. These users can also view public portals. By default, the `Authenticated-User` role is granted minimal privileges, through the following permissions: • `Portal Server: View` • `Portals: Create Portals` • `Portal Templates: Create Portal Templates` • `Pages: Create Pages` • `People Connections: Update People Connections Data` • `People Connections: Connect with People` The `Authenticated-User` role also has permissions to create portals and portal templates. This role inherits permissions from the `Public-User` role. All custom application roles inherit permissions from the `Authenticated-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. By default, the `Public-User` role is granted minimal privileges, that is, only the `Portal Server: View` permission. In the WebCenter Portal, the `Public-User` role is equivalent to `anonymous-role`—a standard OPSS (Oracle Platform Security Services) role. **Caution:** Take care when granting permissions to the `Public-User` role. Avoid granting administrative permissions such as `Portal Server: Manage All`, `Portal Server: Manage Configuration`, or any permission that might be considered unnecessary. See also About Application Permissions. If you do not want unauthenticated users to see WebCenter Portal content that is marked 'public', do not grant the `Portal Server: View` 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 Customizing System Pages for All Portals.	Yes

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 Defining Application Roles.

About Application Permissions

Every application role has specific, defined capabilities known as permissions. These permissions allow users to perform specific actions in the Home portal. Permissions are categorized are listed individually in the subsequent tables:

- Table 38-3 lists the available application permissions in WebCenter Portal.
- Table 38-4 lists the application roles and default permissions assigned to these roles in WebCenter Portal.

No permission, except for `Manage All`, inherits privileges from other permissions.

Understanding Application Permissions

Table 38-3 lists the application-level permissions available in WebCenter Portal.

Table 38-3 Application Permissions

Category	Application Permissions
Portal Server	**Manage All** - Enables access to all *WebCenter Portal Administration* pages: Settings, Portals, Shared Assets, Attributes, andPortal Templates. 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 `Portals: Manage Security and Configuration` 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 `Portal Server: Manage All` permission but excludes security privileges. Users with this permission cannot access the Security page. **View** - Enables users to view WebCenter Portal, and gives them access to the Home portal. See Table 38-2. **Deploy** - Enables users to deploy and propagate a portal. For more information, see Deploying Portals, Templates, Assets, and Extensions.

Table 38-3 (Cont.) Application Permissions

Category	Application Permissions
Portals	**Manage Security and Configuration** - Enables access to all portal administration pages (Overview, Settings, Attributes, Security, Tools and Services), except Assets. Through these pages users can manage portal membership, assign permissions and roles, manage, delete, and deploy and export portals and resources, set portal properties, and manage service availability. • To access portal pages, page and asset permissions must be granted. • To access portal assets, asset permissions must be granted. Includes `Manage Configuration` and `Manage Membership` permissions. **Manage Configuration** - Same as the `Manage Security and Configuration` permission but excludes security privileges. Users with this permission cannot access the Security pages unless they are a portal manager. Users with this permission cannot access the Roles and Members pages. • To access portal pages, page and asset permissions must be granted. • To access portal assets, asset permissions must be granted. Users with this permission must be allowed to view the portal. **Manage Membership** - Enables access to the Roles and Members pages in the portal administration settings. On these pages, users can create, edit, and delete members and roles for the portal. **Create Portals** - Enables users to create portals. See Managing Roles and Permissions for a Portal in *Building Portals with Oracle WebCenter Portal*.
Portal Templates	**Manage All** - Enables users to manage any portal template (through the Portal Templates page) and delete templates accessible to them. See Managing All Portal Templates in *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 the Home portal. **Delete Pages** - Enables users to delete pages in the 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 a new personal page in the Home portal. **Contribute Page Content** - These permissions apply to pages in the Home portal. The permissions do not apply to pages that are created within a portal. Page permissions within a portal are granted by the portal manager. See Managing Roles and Permissions for a Portal in *Building Portals with Oracle WebCenter Portal*.
Application Integration Visualization Templates	**Create, Edit, and Delete Visualization Templates** - Enables users to create, edit and delete visualization templates through WebCenter Portal. **Create Visualization Templates** - Enables users to create visualization templates for the application. **Edit Visualization Templates** - Enables users to edit application-level visualization templates. See Working with Visualization Templates in *Building Portals with Oracle WebCenter Portal*.

Table 38-3 (Cont.) Application Permissions

Category	Application Permissions
Content Presenter Templates	**Create, Edit, and Delete Content Presenter Templates** - Enables users to upload, edit and delete content display templates through WebCenter Portal. **Create Content Presenter Templates** - Enables users to upload content display templates for the application. **Edit Content Presenter Templates** - Enables users to edit application-level content display templates. See Publishing Content Using Content Presenter in *Building Portals with Oracle WebCenter Portal*.
Data Controls	**Create, Edit, and Delete Data Controls** - Enables users to create, edit and delete data controls through WebCenter Portal. **Create Data Controls** - Enables users to create data controls for the application. **Edit Data Controls** - Enables users to edit application-level data controls. See Working with Web Service Data Controls in *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 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.
Page Styles	**Create, Edit, and Delete Page Styles** - Enables users to create, edit, and delete page styles through WebCenter Portal. **Create Page Styles** - Enables users to create page styles for the application. **Edit Page Styles** - Enables users to edit application-level page styles. See Working with Page Styles in *Building Portals with Oracle WebCenter Portal*.
Page Templates	**Create, Edit, and Delete Page Templates** - Enables users to create, edit, and delete page templates through WebCenter Portal. **Create Page Templates** - Enables users to create page templates for the application. **Edit Page Templates** - Enables users to edit application-level page templates. See Working with Page Templates in *Building Portals with Oracle WebCenter Portal*.
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 through WebCenter Portal. **Create Resource Catalogs** - Enables users to create resource catalogs for the application. **Edit Resource Catalogs** - Enables users to edit application-level resource catalogs. See Working with Resource Catalogs in *Building Portals with Oracle WebCenter Portal*.

Table 38-3 (Cont.) Application Permissions

Category	Application Permissions
Skins	**Create, Edit, and Delete Skins** - Enables users to create, edit, and delete skins through WebCenter Portal. **Create Skins** - Enables users to create skins for the application. **Edit Skins** - Enables users to edit application-level skins. See Working with Skins in *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 through WebCenter Portal. **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 Publishing Content Using Content Presenter in *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 WebCenter Portal. **Create Task Flows** - Enables users to create task flows for the application. **Edit Task Flows** - Enables users to edit application-level task flows. See Working with Task Flows in *Building Portals with Oracle WebCenter Portal*.

Default Application Permissions Assignments to Application Roles

Table 38-4 shows the default permissions assigned to built-in application roles.

 - Shows an explicitly granted permission or action.

 - Shows an implied permission because of an explicitly granted permission.

Table 38-4 Default Application Roles and Permissions in WebCenter Portal

Permissions	Administrator	Application Specialist	Portal Creator	Public-User	Authenticated-User
Portal Server Manage All					
Manage Configuration					
View					
Deploy					
Portals Manage Security and Configuration					
Manage Configuration					
Manage Membership					
Create Portals					
Portal Templates Manage All					
Create Portal Templates					

Table 38-4 (Cont.) Default Application Roles and Permissions in WebCenter Portal

Permissions	Administrator	Application Specialist	Portal Creator	Public-User	Authenticated-User
Pages Create, Edit, and Delete Pages and Contribute Content					
Delete Pages					
Edit Pages					
Customize Pages					
View Pages					
Create Pages					
Application Integration Visualization Manage Application Integration Visualization					
Content Presenter Templates Create Content Presenter Templates					
Create, Edit, and Delete Content Presenter Templates					
Edit Content Presenter Templates					
Data Controls Create Data Controls					
Create, Edit, and Delete Data Controls					
Edit Data Controls					
Discussions Create, Edit, and Delete Discussions					
Links Create Links					
Create and Delete Links					
Edit Links					
Page Styles Create Page Styles					
Create, Edit, and Delete Page Styles					
Edit Page Styles					
Page Templates Create Page Templates					
Create, Edit, and Delete Page Templates					

Table 38-4 (Cont.) Default Application Roles and Permissions in WebCenter Portal

Permissions	Administrator	Application Specialist	Portal Creator	Public-User	Authenticated-User
Edit Page Templates					
People Connections Manage People Connections					
Update People Connections Data					
Connect with People					
Resource Catalogs Create Resource Catalogs					
Create, Edit, and Delete Resource Catalogs					
Edit Resource Catalogs					
Skins Create Skins					
Create, Edit, and Delete Skins					
Edit Skins					
Task Flow Styles Create Task Flow Styles					
Create, Edit, and Delete Task Flow Styles					
Edit Task Flow Styles					
Task Flow Styles Create Task Flows					
Create, Edit, and Delete Task Flows					
Edit Task Flows					

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 38-5 and Table 38-6.

Table 38-5 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 38-6 Discussions Server Roles and Permissions - For a Portal

Discussion Server Role	Discussion Server Permissions	WebCenter Portal Equivalent Permissions in a Portal
Portal Manager	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.
Portal Manager	Create Message Create Announcement	• `Discussions-Create and Edit` Create and edit topics. • `Announcements-Create and Edit` Create and edit announcements.
Portal Manager	Read Forum Create Thread	• `Discussions-Reply To` Reply to discussion topics.
Portal Manager	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.

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, Troubleshooting Issues with Users and Roles.

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 more information, see Administering Security in a Portal in *Building Portals with Oracle WebCenter Portal*.

Managing Users

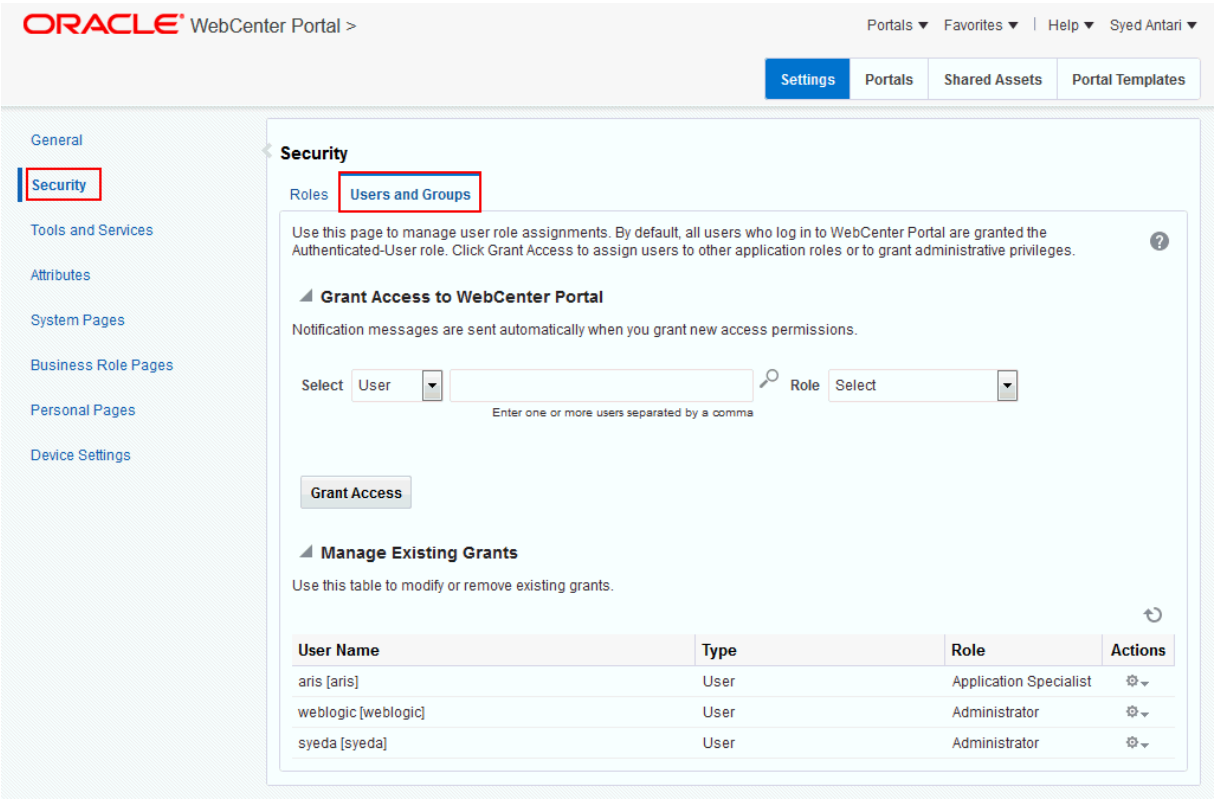
System administrators must ensure that all WebCenter Portal users have appropriate permissions. To get permissions, users must be assigned to an appropriate application role.

System 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 the **Users and Groups** page, you can assign users and groups to roles, change user role assignments, and revoke roles.

To access the **Users and Groups** page, open WebCenter Portal Administration Settings and click **Security**. See Accessing the Settings Pages in WebCenter Portal Administration.

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 Default Application Roles.

Figure 38-3 WebCenter Portal Administration: Users and Groups Page



This section describes how to assign roles and contains the following subsections:

- Adding and Removing Users
- Assigning Users (and Groups) to Application Roles
- Assigning a User to a Different Application Role
- Revoking Application Roles

Adding and 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 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, public 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 Enabling Self-Registration.

Assigning Users (and Groups) to Application 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 Troubleshooting Issues with Users and Roles.

To assign a user (or a group of users) to a different application role:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Security**.

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

   `http://`*`host`*`:`*`port`*`/webcenter/portal/admin/settings/security`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click **Users and Groups** (Figure 38-3).

   This page lists users to whom 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 a group of users.

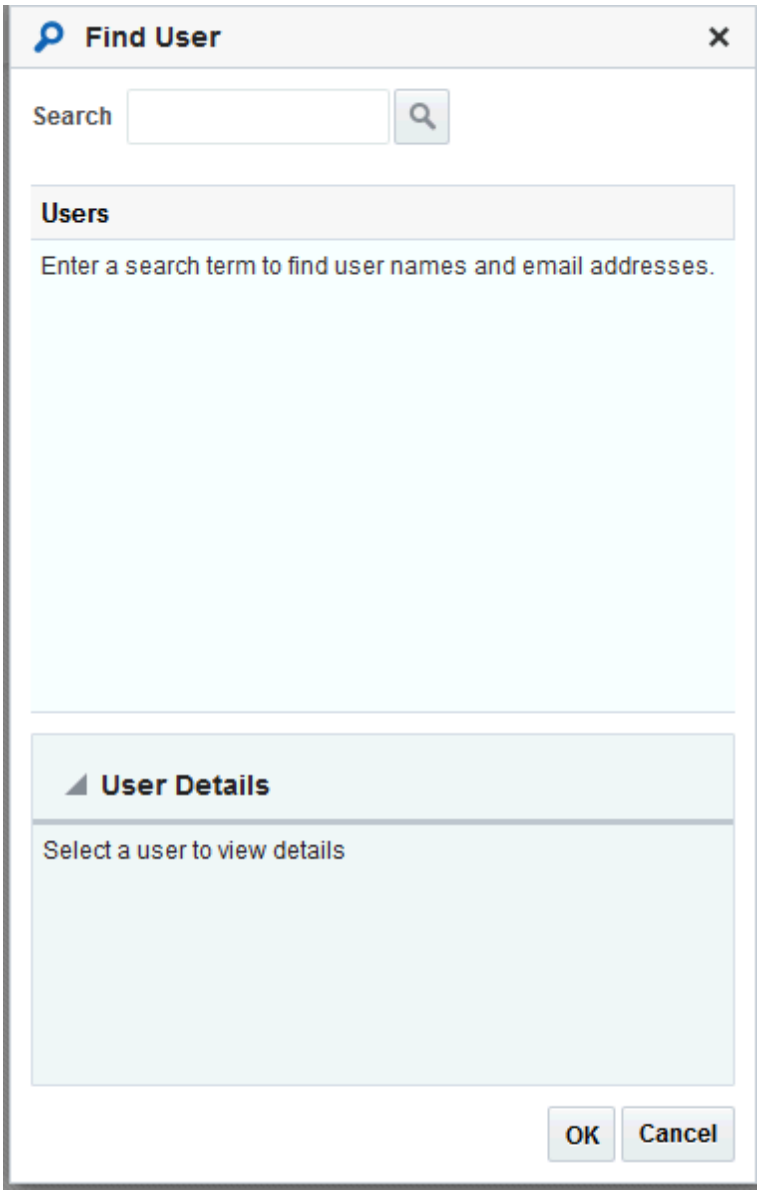
4. If you know the exact name of the user or group, enter the name in the text box, separating multiple names with commas.

   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 38-4).

Figure 38-4 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 Searching for a User or Group in the Identity Store in *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 Searching for a User or Group in the Identity Store in *Building Portals with Oracle WebCenter Portal*

> **Tips:**
>
> - 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).

> **Note:**
>
> Choose **Administrator** only if you want 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 Defining Application Roles).
- When no role is selected, the user assumes the `Authenticated-User` role. See 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.

Assigning a User to a Different Application 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*. You can also assign a user to more than one role.

> **Note:**
>
> You cannot modify your own role or the system administrator's role.
>
> See About Application Roles.

To assign a user to a different role:

1. On the **Settings** page, click **Security**.

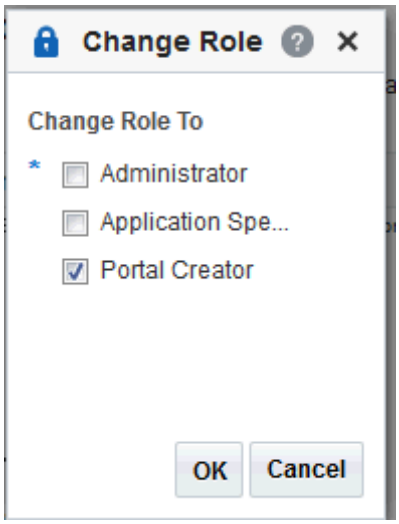
   You can also enter the following URL in your browser to navigate directly to the **Security** page:

   `http://host:port/webcenter/portal/admin/settings/security`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click **Users and Groups** .
3. In the **Manage Existing Grants** table, scroll down to the user whose role assignment you want to modify. Only users with non-default role assignments are listed in the table.
4. Click the **Actions** icon, then select **Change Role** from the drop-down list to open the Change Role dialog.

Figure 38-5 Changing a User's Application Role



5. Select roles as follows:
   - Select **Administrator** only to assign full, administrative privileges for WebCenter Portal.

     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, such as editing the login page, the self-registration page, and profile gallery pages.

See also Default Application Roles.

- Select one or more roles from the list. At least one role must be selected.

  If the role you want is not listed, create a new role that meets your requirements (see Defining Application Roles).

6. Click **OK**.

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 About Application Roles.

To revoke application roles:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Security**.

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

   `http://host:port/webcenter/portal/admin/settings/security`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click **Users and Groups**.

   This page lists users to which additional roles are defined.

3. In the **Manage Existing Grants** table, scroll down to the user from whom you want to revoke roles.

4. Click the **Actions** icon:

   - Select **Change Role**, and deselect the application roles to revoke.
   - Select **Delete Role Assignments** to revoke all roles assigned to that user, and then click **Delete** to confirm.

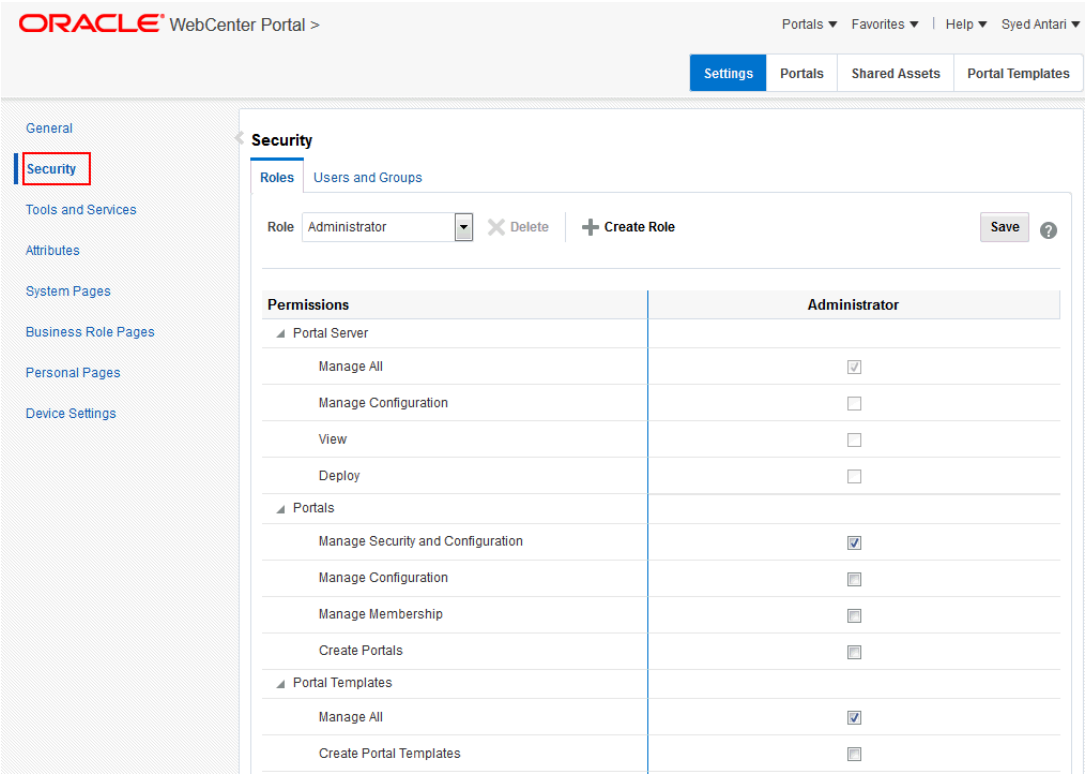
   Access for that user is revoked immediately.

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.

Managing Application Roles and Permissions

WebCenter Portal uses application roles to manage permissions for users working in the *Home portal*. Administrators manage application roles and permissions on the Roles page (Figure 38-6). See Table 38-4 for more information about built-in application roles and permissions.

Figure 38-6 WebCenter Portal Administration: Roles Page



This section explains how to manage application roles and their permissions in WebCenter Portal Administration. It contains the following subsections:

- Viewing Application Roles and Permissions
- Defining Application Roles
- Modifying Application Role Permissions
- Deleting Application Roles

Viewing Application Roles and Permissions

On the Roles page, use the **Roles** drop-down to select an application role and view its associated permissions.

To view permissions associated with a role:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Security**.

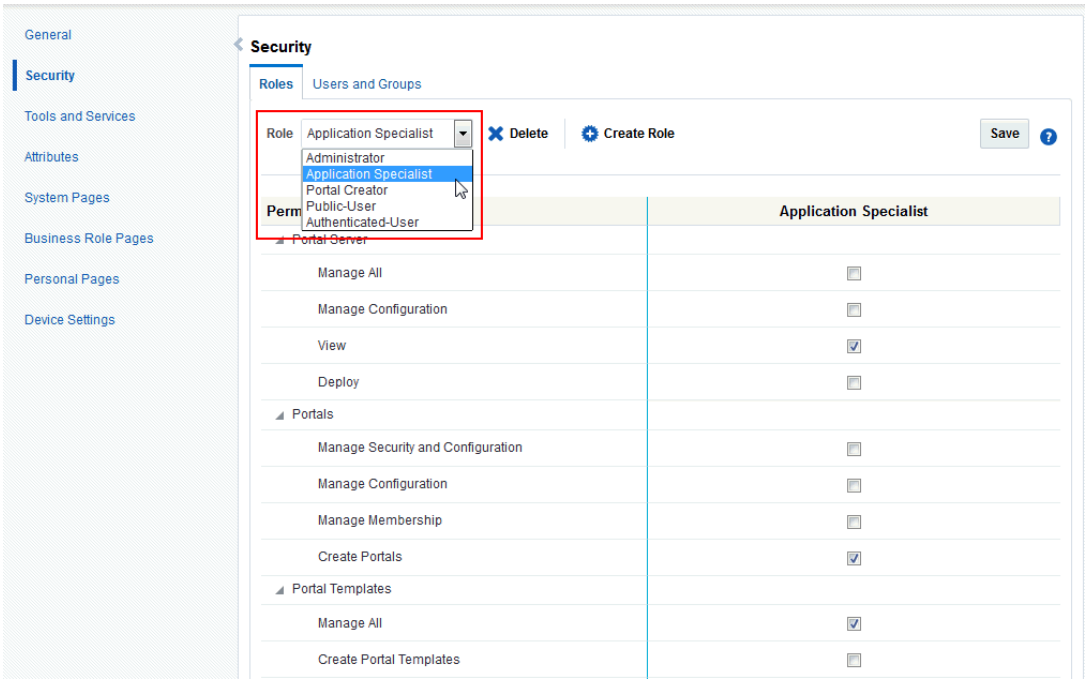
   You can also enter the following URL in your browser to navigate directly to the **Security** page:

   `http://host:port/webcenter/portal/admin/settings/security`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click the **Roles** tab to open the **Roles** page (Figure 38-6), showing the `Administrator` role and its associated permissions by default.
3. From the **Role** drop-down, select a role to view its associated permissions.

Figure 38-7 WebCenter Portal Administration: Roles Page



4. Do any of the following:
   - To create a new application role, see Defining Application Roles.
   - To change defined permissions for a role, see Modifying Application Role Permissions.
   - To delete an application role, see Deleting Application Roles.
5. Click **Save**.

Defining Application Roles

Use roles to characterize groups of WebCenter Portal users to determine what they can see and do in the Home portal and control access to WebCenter Portal administration pages.

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 may fall into multiple roles.

To define a new application role:

1. On the **Settings** page, click **Security**.

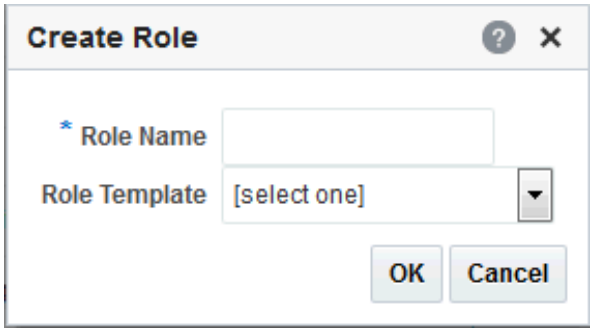
   You can also enter the following URL in your browser to navigate directly to the **Security** page:

   `http://host:port/webcenter/portal/admin/settings/security`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click the **Roles** tab to open the Roles page, showing the `Administrator` role and its associated permissions by default.
3. To define a new role for WebCenter Portal users, click **Create Role** to open the Create Role dialog.

Figure 38-8 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 in the **Role** drop-down. The permissions list shows which actions users with this role can perform. Use the **Roles** drop-down to select another role.

7. To modify user permissions for the role, select or clear each permission check box.
8. Click **Save** to save any changes that you make to the role's permissions.

Modifying Application Role Permissions

Administrators can modify the permissions associated with application roles at any time. Application permissions are described in About Application Permissions.

Application role permissions allow individuals to perform specific actions in the 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 Default Application Roles.

To change the permissions assigned to a role:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Security**.

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

   `http://host:port/webcenter/portal/admin/settings/security`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click the **Roles** (Figure 38-6) tab.

   The page opens, showing the `Administrator` role and its associated permissions by default.

3. From the **Role** drop-down, select the role whose permissions you want to modify.

   The permissions associated with the selected role appear next to the **Permissions** column.

4. Select or deselect **Permissions** check boxes to enable or disable permissions for the role.

   For the built-in roles, be cautious about changing permissions. See Table 38-2.

5. Click **Save**.

The new permissions are effective immediately.

Granting Permissions to the Public-User

Anyone who is not logged in to WebCenter Portal assumes the `Public-User` role. By default, the `Public-User` role is granted minimal privileges, that is, only the `Portal Server: View` permission.

> **Caution:**
>
> Take care when granting permissions to the `Public-User` role. Avoid granting administrative permissions such as `Portal Server: Manage All`, `Portal Server: Manage Configuration`, or any permission that might be considered unnecessary. See also About Application Permissions.

Granting the Portal Server-View Permission

The `Portal Server: View` 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 `Portal Server: View` permission is granted to the `Public-User` role:

- Make sure 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 log in. See Customizing System Pages.

If you do not want unauthenticated users to see WebCenter Portal content that is marked 'public', do not grant the `Portal Server: View` 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 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.

Granting Permissions to the Authenticated-User

Anyone who is logged in to WebCenter Portal assumes the `Authenticated-User` role. By default, the `Authenticated-User` role is granted minimal privileges, through the following permissions:

- `Portal Server: View`

- `Portals: Create Portals`
- `Portal Templates: Create Portal Templates`
- `Pages: Create Pages`
- `People Connections: Update People Connections Data`
- `People Connections: Connect with People`

Other important notes:

- The `Authenticated-User` role always inherits permissions from the `Public-User` role.
- All custom application roles inherit permissions from the `Authenticated-User` role.

Granting Permissions to the Portal Creator

The `Portal Creator` role is given to a logged in user for specifically creating portals.

Out-of-the-box, this role has minimal privileges, through the following permissions: `Portal Server: View` and `Portals: Create Portals`. After creating a portal, the `Portal Creator` role assumes the permissions inherent in the `Portal Manager` role.

Deleting Application Roles

When an application role is no longer required, it is recommended that you remove it. This helps maintain a valid and manageable role list, and prevents inappropriate role assignments.

Application roles can be 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:**
>
> The default application roles of `Administrator`, `Public-User`, and `Authenticated-User` cannot be deleted (the `Application Specialist` and `Portal Creator` roles can be deleted). See Default Application Roles.

To delete an application role:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Security**.

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

   `http://host:port/webcenter/portal/admin/settings/security`

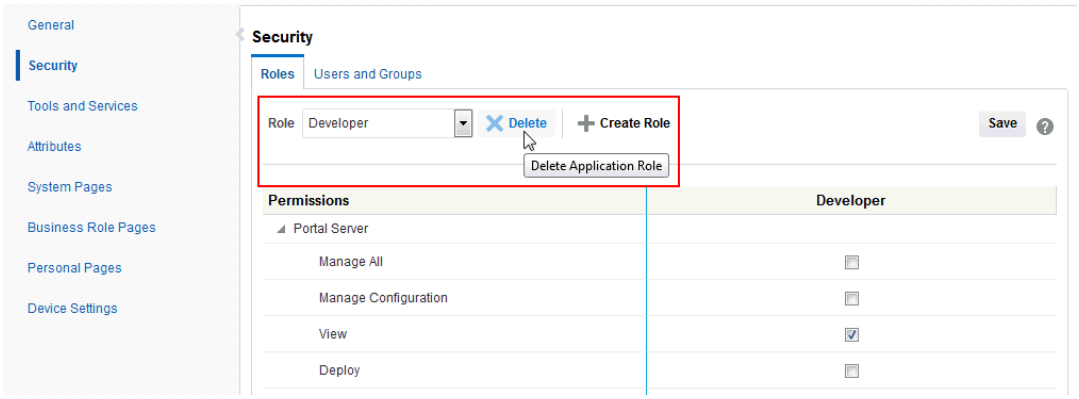
> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click the **Roles** (Figure 38-6) tab.

   The page opens, showing the `Administrator` role and its associated permissions, by default.

3. From the **Role** drop-down, select the role you want to delete, and click **Delete**. Click **Delete** again in the confirmation prompt.

Figure 38-9 Deleting an Application Role



> **Note:**
>
> The default application roles of `Administrator`, `Public-User`, and `Authenticated-User` cannot be deleted (the `Application Specialist` and `Portal Creator` roles can be deleted).

The role is removed from the table. Any users that were assigned to this role assume the default `Authenticated-User` role.

39

Working with Global Attributes Across Portals

Use the **Attributes** page in WebCenter Portal Administration to manage global attributes, which can be used by any portal in WebCenter Portal.

> **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 All` or `Portal Server: Manage Configuration`
>
> For more information about permissions, see About Application Roles and Permissions.

Topics:

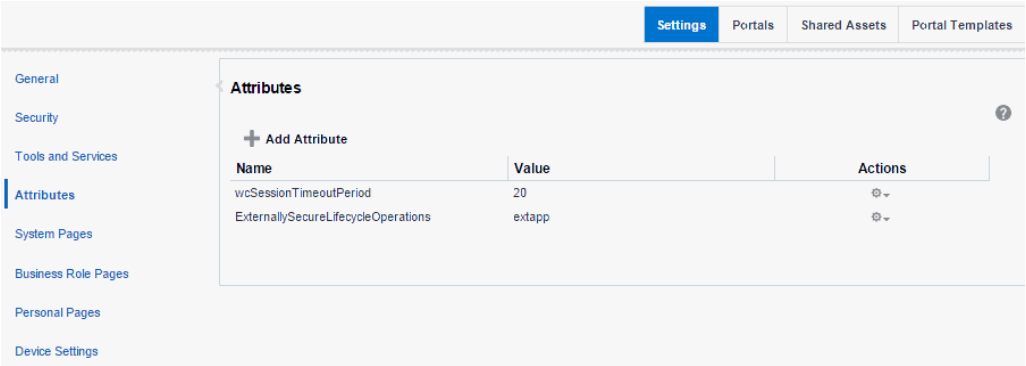
- About Global Attributes
- Adding a Global Attribute
- Editing a Global Attribute
- Deleting a Global Attribute

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 managers 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 Working with Portal Attributes in *Building Portals with Oracle WebCenter Portal*.

In addition to portal-specific attributes, system administrators can add and manage global attributes from the **Attributes** page in WebCenter Portal Administration. Global attributes are available for use by any portal.

Figure 39-1 WebCenter Portal Administration: Attributes



Custom attributes of a private portal are internal and cannot be accessed by a user who does not have access to the portal. For custom attributes to be accessible either the portal should be public or the user must have access to the private portal. Otherwise, you can include global attributes in your portal that are available for use by any portal.

A custom attribute is simply a name value pair (such as `customerId=400`, `orderId=11`, or `userName=Smith`). 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 Expression Language Expressions in *Developing for Oracle WebCenter Portal*.

Adding a Global Attribute

To add a new global attribute for use by any portal:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Attributes**.

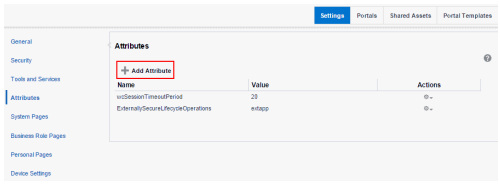
   You can also enter the following URL in your browser to navigate directly to the **Attributes** page:

   `http://host:port/webcenter/portal/admin/attributes`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

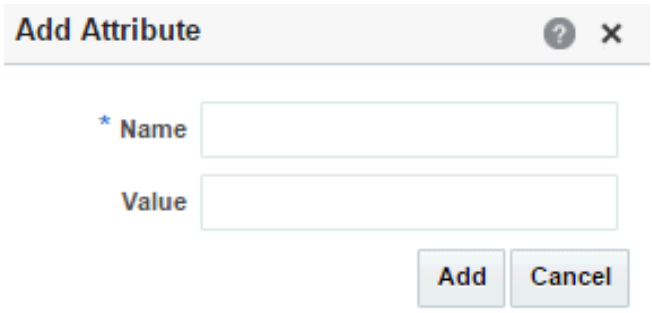
2. On the **Attributes** page, click **Add Attribute** (Figure 39-2).

Figure 39-2 WebCenter Portal Administration: Add Attribute



The Add Attribute dialog opens (Figure 39-3).

Figure 39-3 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. Enter a **Value** for the custom attribute. The value you type is treated as a string value. A value is optional for global attributes.
5. Click **Add** to save the custom attribute.

Editing a Global Attribute

To edit a global attribute:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Attributes**.

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

   `http://host:port/webcenter/portal/admin/attributes`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. On the **Attributes** page, 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.

Deleting a Global Attribute

To delete a global attribute:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Attributes**.

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

   `http://host:port/webcenter/portal/admin/attributes`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. On the **Attributes** page, click the **Actions** icon for the attribute and select **Delete Attribute**.
3. In the confirmation dialog, click **Delete**.

40
Customizing System Pages

Use the **System Pages** page in WebCenter Portal Administration to view and customize the built-in system pages available in WebCenter Portal.

> **Note:**
>
> Any changes made to a system page by the system administrator are reflected in all portals. For example, if the system administrator adds an image to the **Announcements** system page, that image will display in the **Announcements** system page in all portals. It is not possible to modify a system page exclusively for an individual portal.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

> **Permissions:**
>
> To perform the tasks in this chapter, you must have the WebCenter Portal `Administrator` role or a custom role that grants the following permissions:
>
> - `Portal Server: Manage All` or `Portal Server: Manage Configuration`
> - `Pages: Create, Edit, and Delete Pages`
>
> For more information about permissions, see About Application Roles and Permissions.

Topics:

- About System Pages
- Customizing System Pages for All Portals
- Setting System Page Properties
- Removing All Page Customizations from a System Page

About System Pages

System pages are a set of out-of-the-box utility pages that are designed to display in a specific circumstance. For example, users who are not logged in when they visit a portal may see the public **Welcome** page. System pages support rapid deployment of a portal to fulfill a range of immediate 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 customize system pages to reflect your company brand, to provide useful hints, or to provide additional functionality (such as task flows and portlets). Once you customize a system page, that page displays your customizations along with an updated WebCenter Portal look and feel. Until you customize a system page, WebCenter Portal displays the generic version of the system page with the look and feel used in prior releases of WebCenter Portal.

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. In other words, authorized users can add a seeded task flow to this page and then customize it to apply the customizations to all instances of the task flow. Note that custom task flows created through the **Assets** or **Shared Assets** page cannot be edited using this page. For more information, see Customizing Task Flows.

> **Note:**
>
> For a list and description of the system pages, see About Built-In System Pages.

About Built-In System Pages

Table 40-1 lists and describes the system pages that are included with WebCenter Portal and provides information about the context in which they appear.

Table 40-1 Built-In System Pages

Page	Description	Context
About WebCenter Portal	Provides information about WebCenter Portal, including version and copyright information.	Displays when user clicks the **Help** menu, and selects **About**, or click the **About WebCenter Portal** link in the footer.

Table 40-1 (Cont.) Built-In 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 Tracking Portal Activities in *Using Portals in Oracle WebCenter Portal*.	Appears by default in the Home portal for every authenticated (logged-in) user.
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 Tracking Portal Activities in *Using Portals in Oracle WebCenter Portal*.	Appears by default on the Home page in the built-in portal template.
Analytics	Provides information about application usage and performance metrics. For more information, see 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.
Announcements	Displays the Announcement Manager task flow. For more information, see Working with Announcements in *Using Portals in Oracle WebCenter Portal*.	Appears by default in the built-in portal template.
Discussions	Displays the Discussion Forum Manager task flow. For more information, see Viewing and Participating in Discussions in *Using Portals in Oracle WebCenter Portal*.	Appears by default in the built-in portal template.
Documents	Displays the Content Manager 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 Adding and Managing Documents in *Using Portals in Oracle WebCenter Portal*.	Appears by default in the built-in portal template.
Error Encountered	Displays an error page when an error occurs.	Appears when an application error occurs.
Events	Displays the Events task flow. For more information, see Working with Calendars and Events in *Using Portals in Oracle WebCenter Portal*.	Appears by default in the built-in portal template.
Lists	Displays the List Manager task flow. For more information, see Working with Lists in *Using Portals in Oracle WebCenter Portal*.	Appears by default in the built-in portal template.
Members	Provides features for managing the members of a portal. For more information, see Managing Members and Assigning Roles in a Portal in *Building Portals with Oracle WebCenter Portal*.	Appears in the default navigation as the **Members** page in some built-in portal templates.
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.

Table 40-1 (Cont.) Built-In 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.
Page Viewer	Displays an external web site (such as google.com) in a portal, surrounded by the page template. For more information, see Adding Items to the Portal Navigation in*Building Portals with Oracle WebCenter Portal*.	Used when portal navigation contains an External URL item (with target Same Page). When users click on such links in the navigation, the Page Viewer is used.
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.
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 Viewing and Accessing Available Portals in *Using Portals in Oracle WebCenter Portal*.	Appears when users select **Portal Browser** from the **Portals** menu.
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 Working with Portal Templates in*Building Portals with Oracle WebCenter Portal*.	Appears in the administration pages on the **Portal Templates** 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 Managing Your Profile and Creating and Managing Documents in *Using Portals in Oracle WebCenter Portal*.	Appears by default in the Home portal of every authenticated (logged-in) user.
Public Portals	Displays all public portals in the Portal Browser.	Appears when users select View Public Portals on the WebCenter Portal Sign In page.
Resource Viewer	Displays a portlet resource in a portal, surrounded by the page template. For more information, see Adding Items to the Portal Navigation in *Building Portals with Oracle WebCenter Portal*.	Used when portal navigation contains a link to a portlet resource. When users click on such links in the navigation, the Resource Viewer is used.
Search	Displays the Search Results page. For more information, see the About Searching in WebCenter Portal in *Using Portals in Oracle WebCenter Portal*.	Renders dynamically to display the results of a search.
Self-Registration	Provides a means of enabling users to create their own login accounts to your WebCenter Portal. For more information, see Registering Yourself with WebCenter Portal in *Using Portals in Oracle WebCenter Portal*.	Appears when users click **Register** on the WebCenter Portal Sign In page.

Table 40-1 (Cont.) Built-In System Pages

Page	Description	Context
Self-Service Membership	Provides a means of subscribing to a portal that is configured to allow membership by subscription. For more information, see Joining a Portal in *Using Portals in Oracle WebCenter Portal*.	Appears when users initiate a subscription to a portal.
Sign In	Provides fields for logging in to your portal.	Appears instead of the WebCenter Portal Sign In page when you disable public access to all application pages and when your current session expires.
Tag Center	Displays the Tag Center to enable users to manage tags. For more information, see Working with Tags in *Using Portals in Oracle WebCenter Portal*.	Rendered dynamically when users click a tag in a Tags task flow or in search results.
Task Flow Editor	Provides an environment for customizing all instances of a built-in 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 Customizing Task Flows.	Allows authorized users to add built-in task flows and then customize all instances of those task flows.
Task Flow Viewer	Displays a task flow resource in a portal, surrounded by the page template. For more information, see Adding Items to the Portal Navigation in *Building Portals with Oracle WebCenter Portal*	Used when portal navigation contains a link to a task flow resource. When users click on such links in the navigation, the Task Flow Viewer is used.
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.
Unavailable	Displays a message notifying users that the portal is not available.	Appears when users navigate to a portal that is offline.
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.
WebCenter Portal Welcome Page	Displays login fields, a **Register** link (if self-registration is enabled), a link to public portals, footer links, 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.

Customizing System Pages for All Portals

You can customize built-in 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.

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 a system page or system page variant:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **System Pages**.

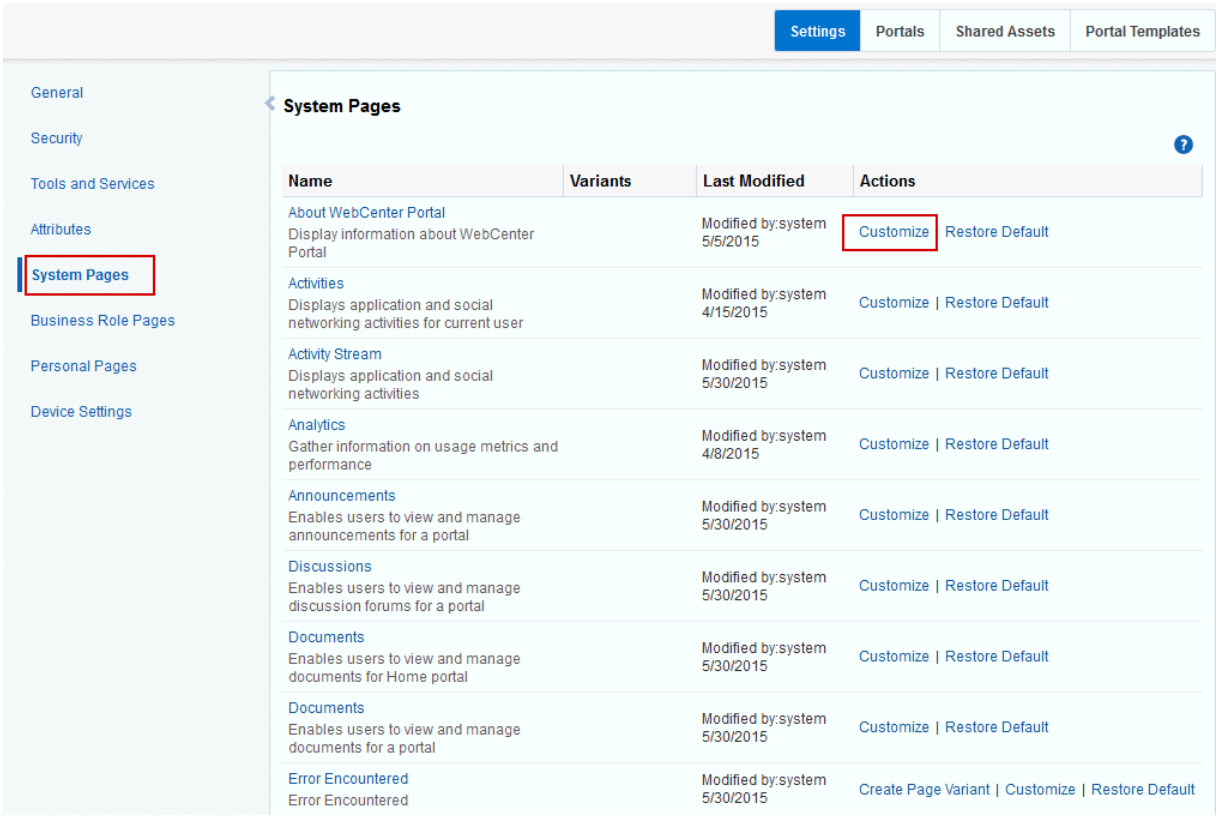
   You can also enter the following URL in your browser to navigate directly to the **System Pages** page:

   `http://host:port/webcenter/portal/admin/settings/systempages`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*

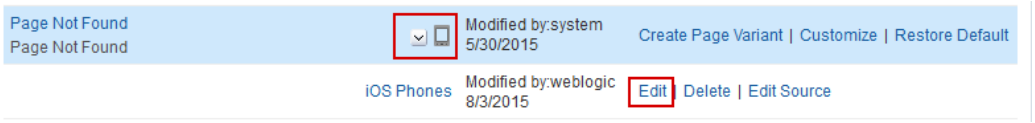
2. Click the **Customize** link next to the system page to open it in Composer.

Figure 40-1 Customize Link Next to a System Page



3. To customize a variant of a system page for a device group, expand the system page variant icon, then click **Edit** for the device group you want to customize.

Figure 40-2 Customizing a System Page Variant for a Device Group



4. Edit and then save the page.

> **See Also:**
>
> For information about editing system pages, see Editing a Page in Structure View in Composer in *Building Portals with Oracle WebCenter Portal*.

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 Administering Device Settings.

For portals that do not use responsive page templates, you can define page variants that optimize the display of a portal page for a specific device group. At runtime, WebCenter Portal can check if a page variant exists for the device being used to display a portal page, and displays the page using the page variant definition. By default, this check is not performed. To turn it on, see Enabling Page Variants for Device Groups.

Only system administrators can create page variants for the application-level system pages. 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 variants 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 **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **System Pages**.

   You can also enter the following URL in your browser to navigate directly to the **System Pages** page:

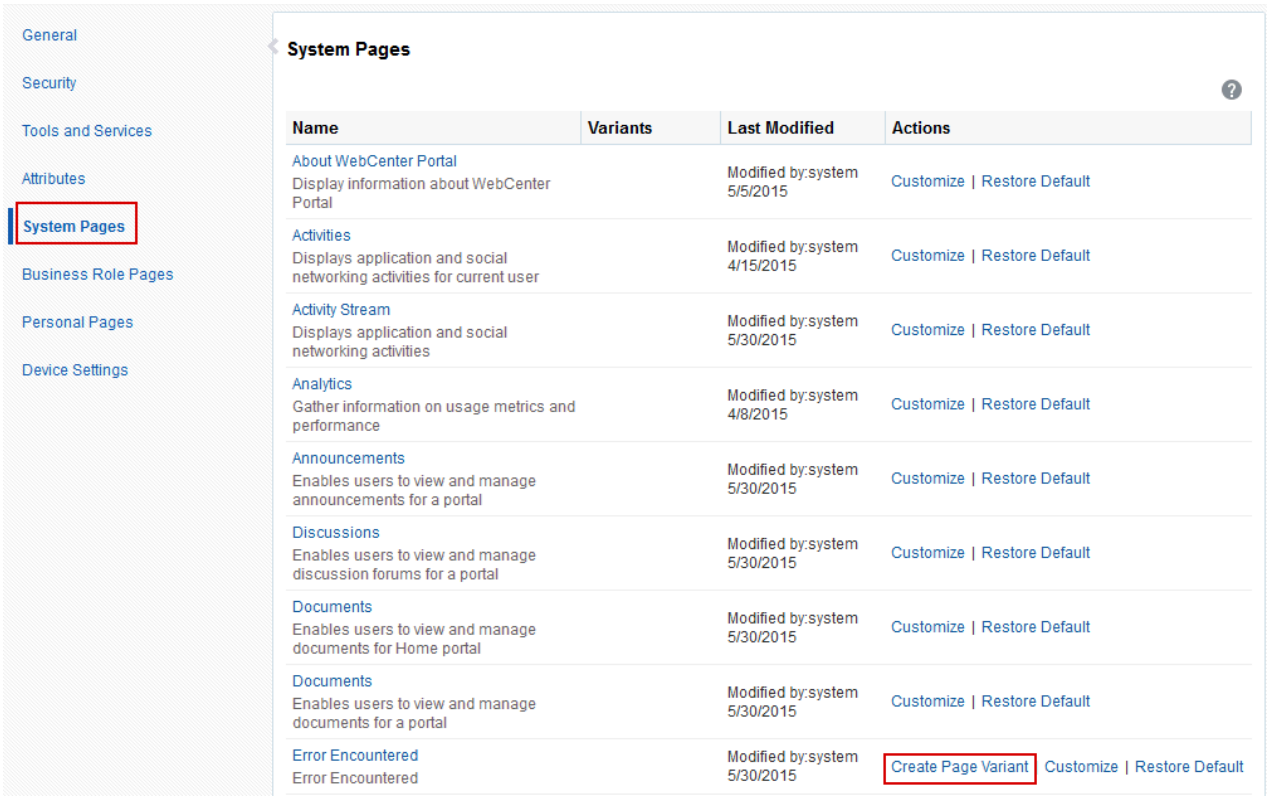
   `http://host:port/webcenter/portal/admin/settings/systempages`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click the **Create Page Variant** link next to the system page for which you want to create a page variant.

> **Note:**
>
> You can create page variants for the following system pages only:
>
> - **Error Encountered**
> - **No Pages Accessible**
> - **Page Not Found**
> - **Portal Not Found**
> - **Self-Registration**
> - **Sign In**
> - **Unauthorized**
> - **Unavailable**
> - **WebCenter Portal Welcome Page**

Figure 40-3 Create Page Variant Link Next to a System Page



3. In the Create Page Variant dialog that opens, select the device group for which you want to create a page variant from the **Device Group** drop-down list.

   The base page is seeded in the system. The base page is always rendered for devices belonging to the default device group.

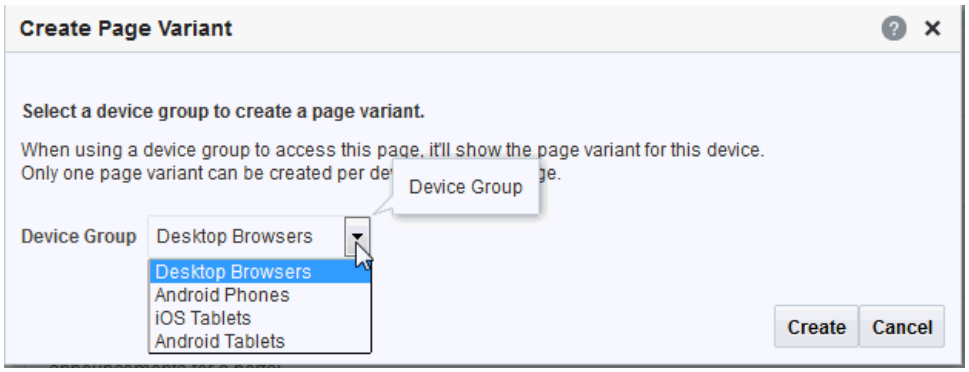
   For more information about the default device group, see Administering Device Settings.

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 Browsers** 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 Browsers** variant is displayed, not the base page. If you do not change the default device group, the **Desktop Browsers** 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.

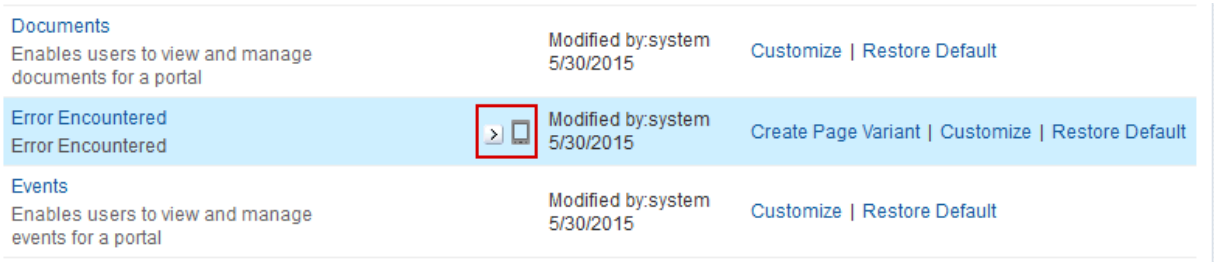
Figure 40-4 Create Page Variant Dialog



4. Click **Create**.

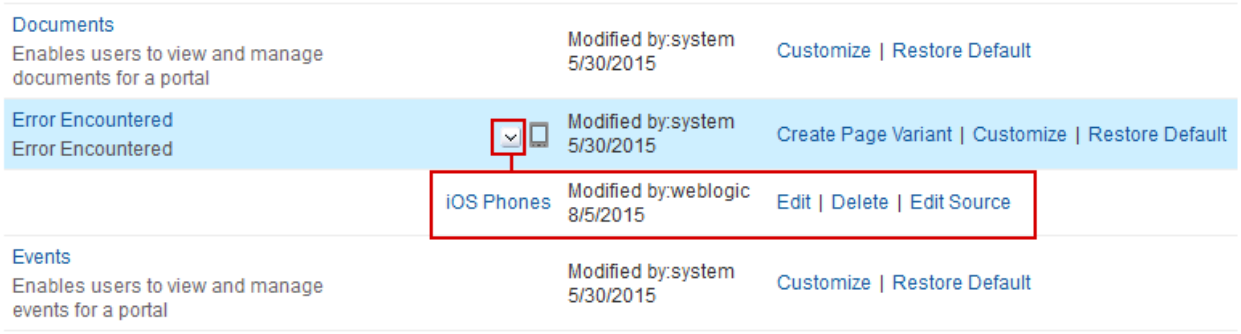
   A mobile icon with an expand button appears next to the page, indicating that a page variant for the page is available.

Figure 40-5 Icon Showing That a Page Variant is Available



5. Click the **Expand** button to view the device group page variant.

Figure 40-6 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 system page in the page editor.

     For information about editing system pages, see Working in Structure View in the Page Editor in *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 Viewing and Modifying Page Source Code in *Building Portals with Oracle WebCenter Portal*.

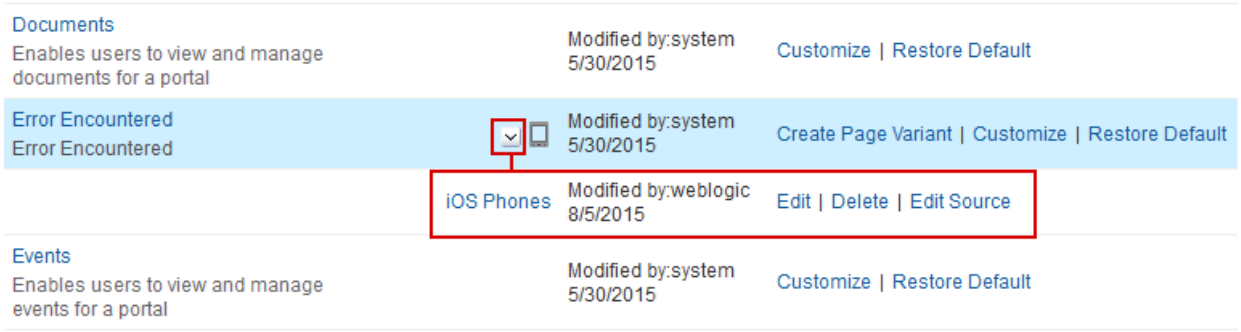
Managing a Page Variant of a System Page for Device Groups

For information about page variants for device groups and creating a page variant for a system page, see 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 40-7).

Figure 40-7 Page Variant for a Device Group



2. To edit the page variant in Composer, click **Edit**.

For information about editing system pages, see Editing a Page in Structure View in Composer in *Building Portals with Oracle WebCenter Portal*.

3. To delete the page variant, click **Delete**.
4. Confirm the deletion by clicking **Delete** again.
5. To edit the source code, click **Edit Source**.

   For information about editing page source code, see Viewing and Modifying Page Source Code in *Building Portals with Oracle WebCenter Portal*.

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 **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **System Pages**.

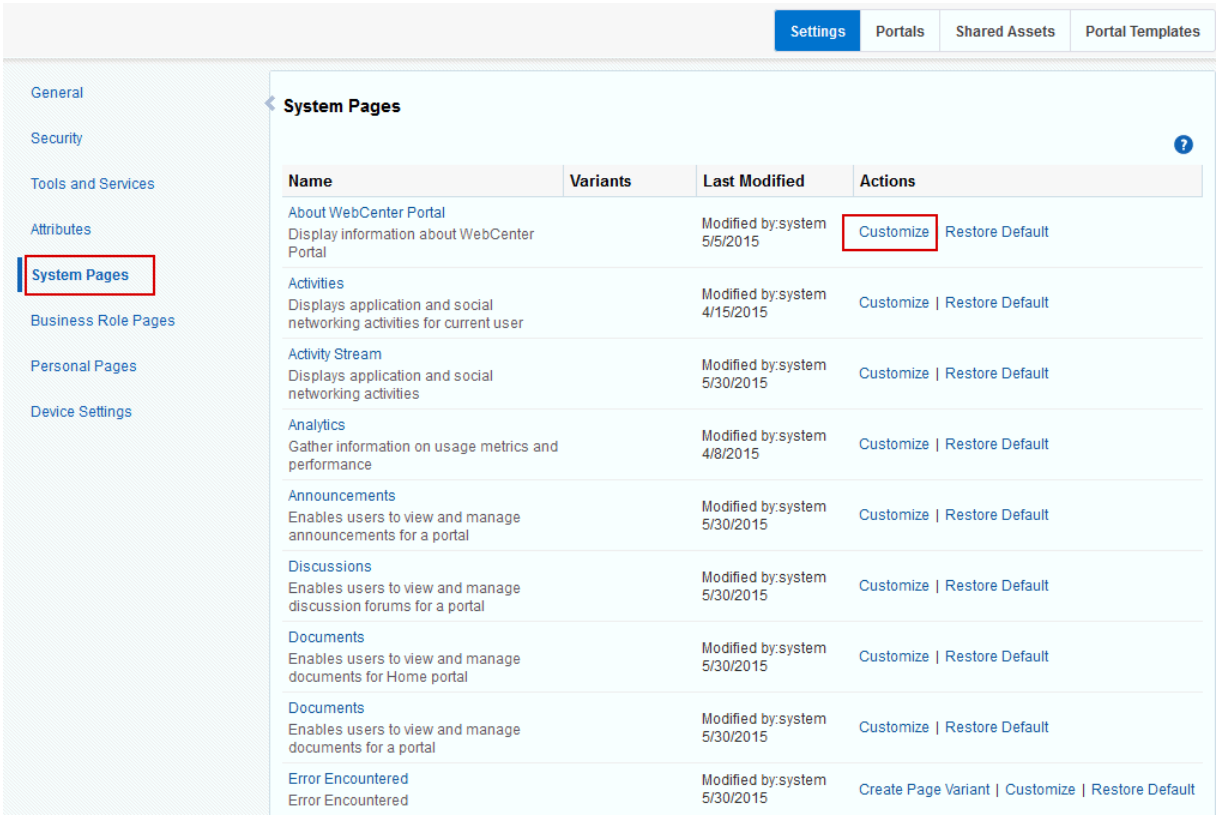
   You can also enter the following URL in your browser to navigate directly to the **System Pages** page:

   `http://host:port/webcenter/portal/admin/settings/systempages`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

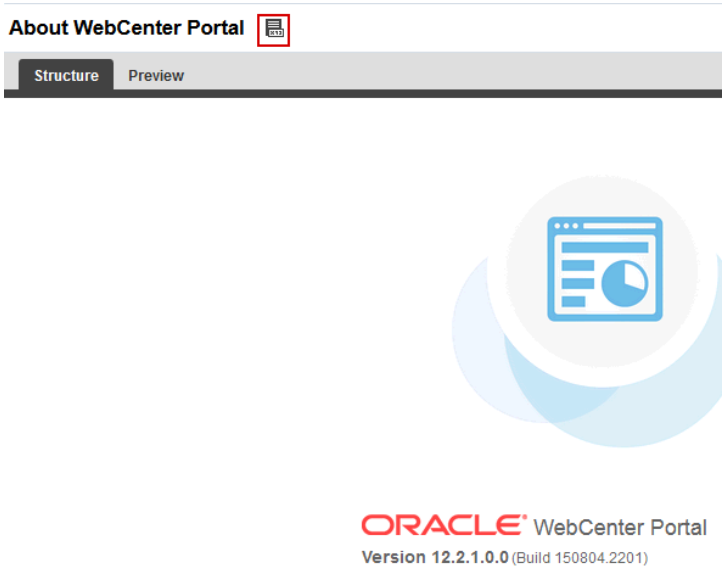
2. Click the **Customize** link next to a system page to open it in Composer.

Figure 40-8 Customize Link Next to a System Page



3. Click the **Page Properties** icon at the top of the page to open the Page Properties dialog.

Figure 40-9 Page Properties Icon



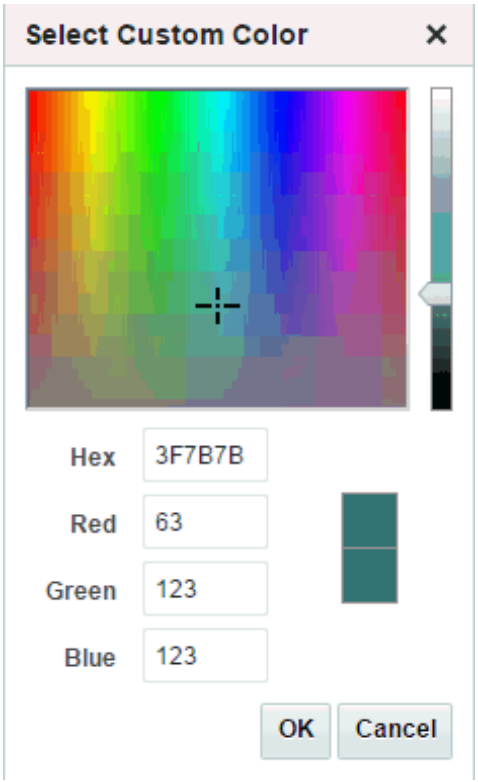
4. On the **Display Options** tab, modify settings as required:
   a. To set the page background color, open the color picker next to **Background Color**, and select a background color for the page in one of the following ways:

- Select a color by clicking it.
- Enter the color's RGB equivalent in the **Background Color** field.

> **Tip:**
>
> Enter RGB values in the format `RRGGBB` or `#RRGGBB` or `r,g,b`

- Create a custom color by clicking **Custom Color** in the picker and selecting a color and a saturation level using the selector and the slider provided.

Figure 40-10 Custom Color Picker



Click **OK** to enter the color value in the **Background Color** field.

b. To set a background image for the page, enter in the **Background Image** field a full URL or a URL relative to the application root. For example:

```
http://www.abc.com/image.jpg
```

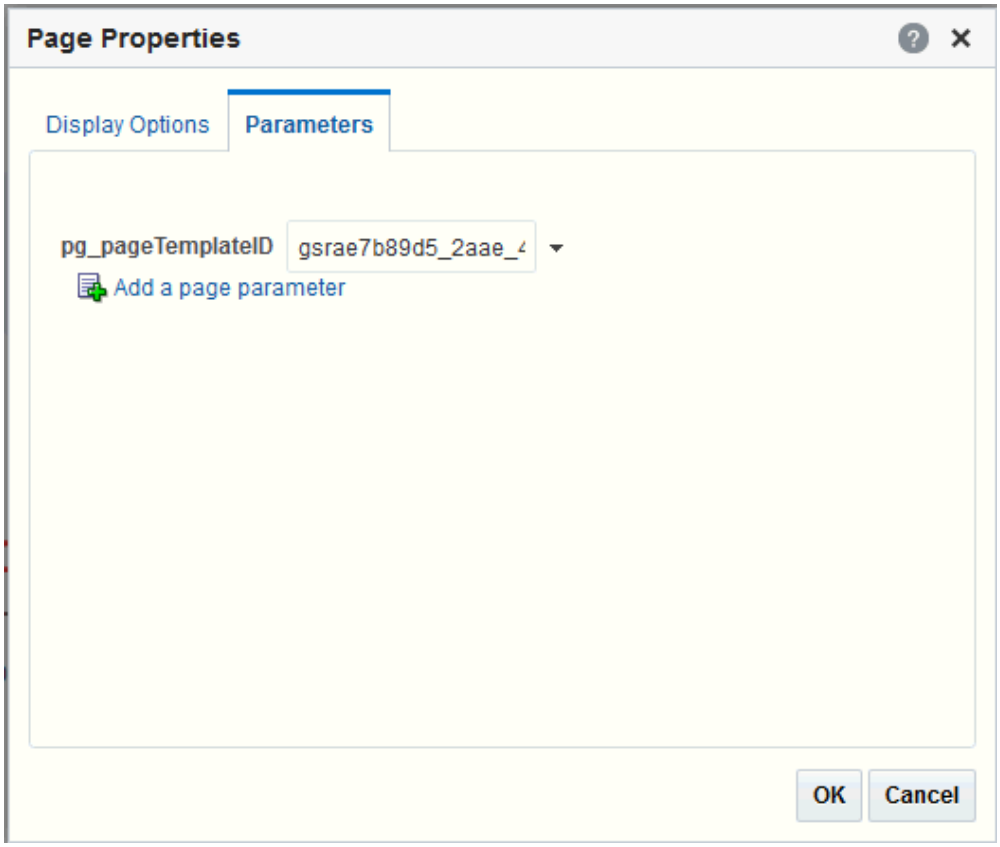
c. In the **Other CSS** field, add any desired CSS encoding that is not covered by the other page properties. Examples:

```
background-position:center;
background: #F8F8FF url(http://www.google.com/intl/en_ALL/images/
logo.gif) no-repeat fixed top; font-size: xx-small
```

You must use standard CSS syntax for this value to be valid. For more information about **Other CSS**, see Other CSS Property: Usage and Examples in *Building Portals with Oracle WebCenter Portal*.

5. On the **Parameters** tab, modify existing parameters as required.

Figure 40-11 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 Expression Language Expressions in *Developing for Oracle WebCenter Portal*.

System pages include a default page parameter: .

- `pg_pageTemplateID`: By default, specifies the system page using the default page template for the system page. To use a custom page template for the system page, enter a GUID value for the new page template.

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.

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 Customizing Task Flows.

To remove all customizations from a system page:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **System Pages**.

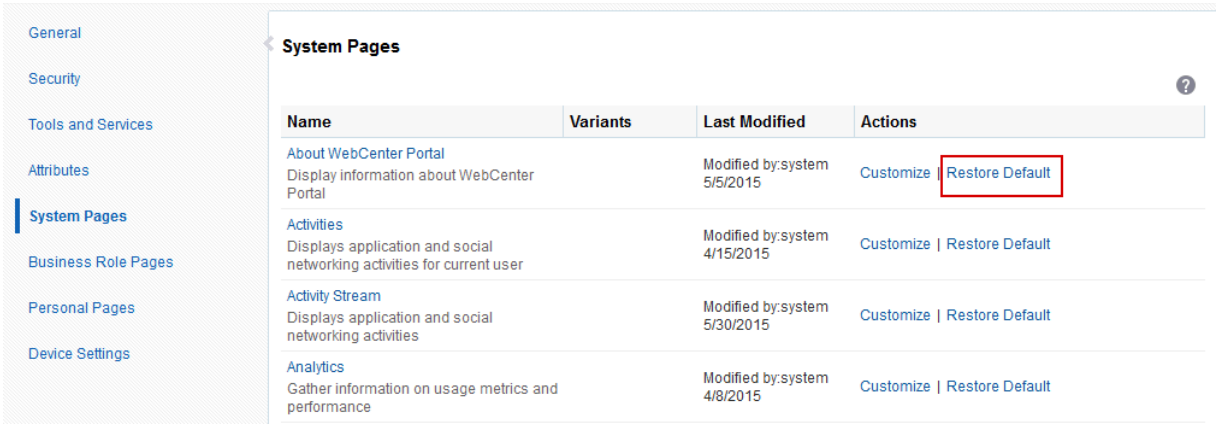
   You can also enter the following URL in your browser to navigate directly to the **System Pages** page:

   `http://host:port/webcenter/portal/admin/settings/systempages`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click the **Restore Default** link next to the system page (Figure 40-12).

Figure 40-12 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.

41

Managing Business Role Pages

Use the **Business Role Pages** page in WebCenter Portal Administration to create and target business role pages and perform other related business role page management 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 permissions:
>
> - `Portal Server: Manage All` or `Portal Server: Manage Configuration`
> - `Pages: Create, Edit, and Delete Pages`
>
> For more information about permissions, see About Application Roles and Permissions.

Topics:

- About Business Role Pages
- Setting Page Creation Defaults for Business Role Pages
- Creating a Business Role Page
- Specifying the Target Audience for a Business Role Page
- Revoking Access to a Custom Business Role Page
- Showing and Hiding Business Role Pages
- Setting a Default Display Order for Business Role Pages
- Editing a Business Role Page
- Editing the Source of a Business Role Page
- Copying a Business Role Page
- Removing All User Customizations from a Business Role Page
- Deleting a Custom Business Role Page

About Business Role Pages

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.

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 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. For a listing of built-in business role pages, see About Built-In Business Role Pages.

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. Built-in business role pages (see Table 41-1) have preconfigured access settings that cannot be altered. For information about how to alter access settings on seeded business role pages, see Setting Access on a Built-in 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 Specifying the Target Audience for a Business Role Page).

Table 41-1 lists and describes the built-in 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** page (for more information, see Customizing System Pages).

About Built-In Business Role Pages

Table 41-1 lists the built-in business role pages in WebCenter Portal.

Table 41-1 Built-In 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 Tracking Portal Activities and Working with Feedback and the Message Board in *Using Portals in 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 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 Working with Task Flows in *Building Portals with Oracle WebCenter Portal*.	Appears by default in the Home portal views of each authenticated user.

Table 41-1 (Cont.) Built-In Business Role Pages

Page	Description	Context
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 Managing Your Profile in *Using Portals in Oracle WebCenter Portal* and Working with Task Flows in *Building Portals with Oracle WebCenter Portal*.	Appears by default in the Home portal views of each authenticated user.
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 to enable users to manage tags. For more information, see Adding Tagging to a Portal in *Using Portals in Oracle WebCenter Portal*.	Rendered dynamically when users click a tag in a Tags task flow or in search results.
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 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 Using WebCenter Portal Impersonation in *Using Portals in Oracle WebCenter Portal*. For information about impersonation ELs and APIs, see Using WebCenter Portal Impersonation ELs and APIs in *Developing for Oracle WebCenter Portal*.	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 Configuring Oracle Access Manager.

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 Setting Page Creation Defaults for Personal Pages in *Using Portals in Oracle WebCenter Portal*). Defaults for pages created in a portal are controlled by the portal manager (for more information, see Creating and Editing a Portal Page in *Building Portals with Oracle WebCenter Portal*).

To set page creation defaults for business role pages:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Business Role Pages**.

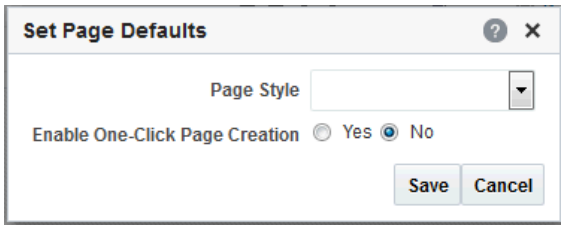
   You can also enter the following URL in your browser to navigate directly to the **Business Role Pages** page:

   `http://host:port/webcenter/portal/admin/settings/businessrolepages`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click **Set Page Defaults** to open the Set Page Defaults dialog (Figure 41-1).

Figure 41-1 Set Page Defaults Dialog



3. Select a page layout from the **Page Style** drop-down list.

   > **See Also:**
   >
   > For an overview of built-in page styles, see Built-In Page Styles in *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 pages using the specified **Page Style**.

> **Tip:**
>
> When you create pages by bypassing the Create Page dialog, the new page has a generic name.

- **No**: Display the Create Page dialog, with the specified **Page Style** selected as the default in the Create Page dialog for all of your pages. You can select a different style for your new pages.

5. Click **Save**.

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 Copying a Personal Page and Copying a Business Role Page.

To create a new business role page:

1. On the **Settings** page, click **Business Role Pages**.

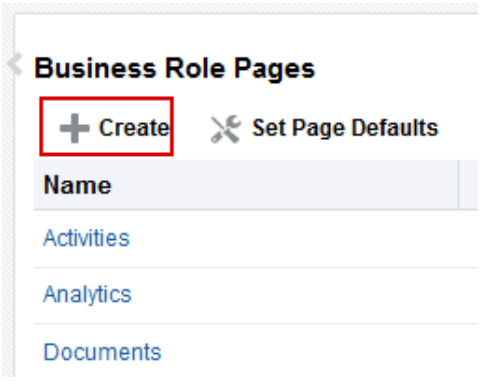
   You can also enter the following URL in your browser to navigate directly to the **Business Role Pages** page:

   `http://host:port/webcenter/portal/admin/settings/businessrolepages`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click **Create**.

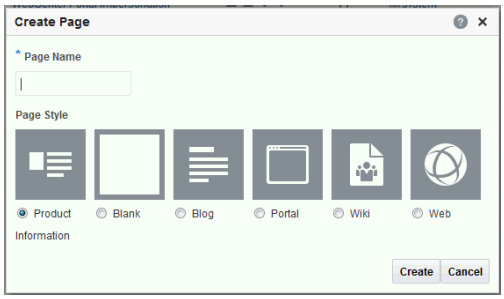
Figure 41-2 Create Option for a Business Role Page



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 41-3 Create Page Dialog



> **See Also:**
>
> For an overview of built-in page styles, see Built-In Page Styles in *Building Portals with Oracle WebCenter Portal*

4. Click **Create**.

   The new page appears in the list of business role pages.

> **Note:**
>
> 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 in *Building Portals with Oracle WebCenter Portal*.

> **Note:**
>
> Make sure that the page style you select does not contain any content publishing components. Content publishing is not supported in Business Role pages. Built-in page styles that you can use are Product Information, Blank, Blog, Portal, Wiki, and Web Information.

5. Next steps:
   - Define the page audience, as described in Specifying the Target Audience for a Business Role Page.
   - Choose the page display order, as described in Setting a Default Display Order for Business Role Pages.

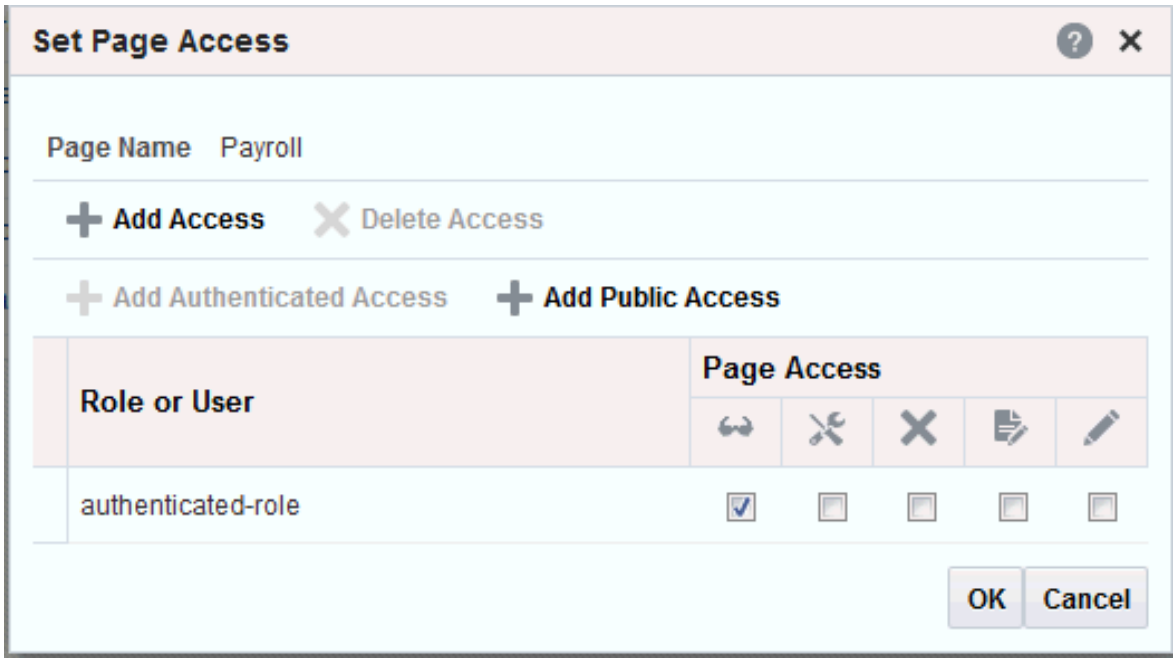
Specifying the Target Audience for a Business Role Page

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 Pages` permission, 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 Setting Access on a Custom Business Role Page).
>
> You cannot alter the default access settings of seeded business role pages (see Table 41-1) through the WebCenter Portal user interface. For information about how to set access on seeded business role pages, see Setting Access on a Built-in Business Role Page.

You can find controls for setting page access on a business role page that you create in WebCenter Portal Administration (Figure 41-4).

Figure 41-4 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:

- Setting Access on a Custom Business Role Page
- Providing Public Access to a Custom Business Role Page
- Setting Access on a Built-in Business Role Page

Setting Access on a Custom Business Role Page

As the system administrator, you can set access on the business role pages that you create. However, you cannot use the WebCenter Portal Administration user interface to set access on built-in business role pages (see Setting Access on a Built-in Business Role Page).

To specify the target audience for a custom business role page that you created:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Business Role Pages**.

   You can also enter the following URL in your browser to navigate directly to the **Business Role Pages** page:

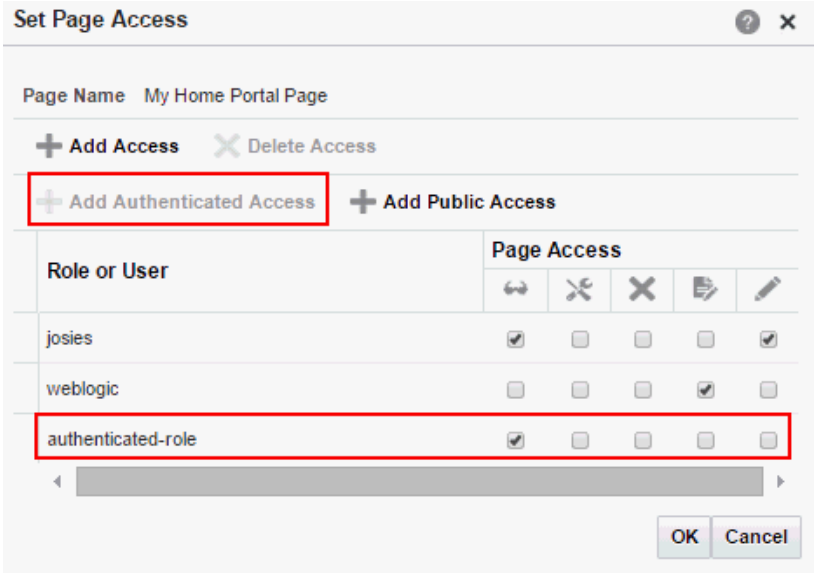
   `http://host:port/webcenter/portal/admin/settings/businessrolepages`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click the **Actions** icon for the page you want to secure, and select **Set Page Access** to open the Set Page Access dialog.
3. To grant page access permissions to all authenticated users (that is, to users who are logged in to WebCenter Portal), click **Add Authenticated Access**.

   The role `authenticated-role` is added under **Role or User** with default **View** access to the page.

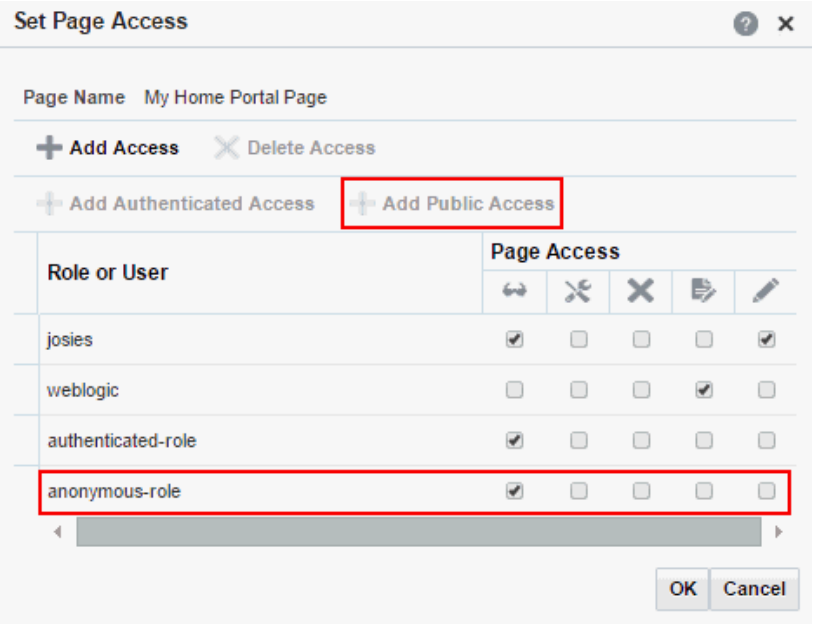
Figure 41-5 Authenticated Role Access



4. To grant page access permissions to all public users (that is, users who have not logged in to WebCenter Portal as well as those who have) click **Add Public Access**.

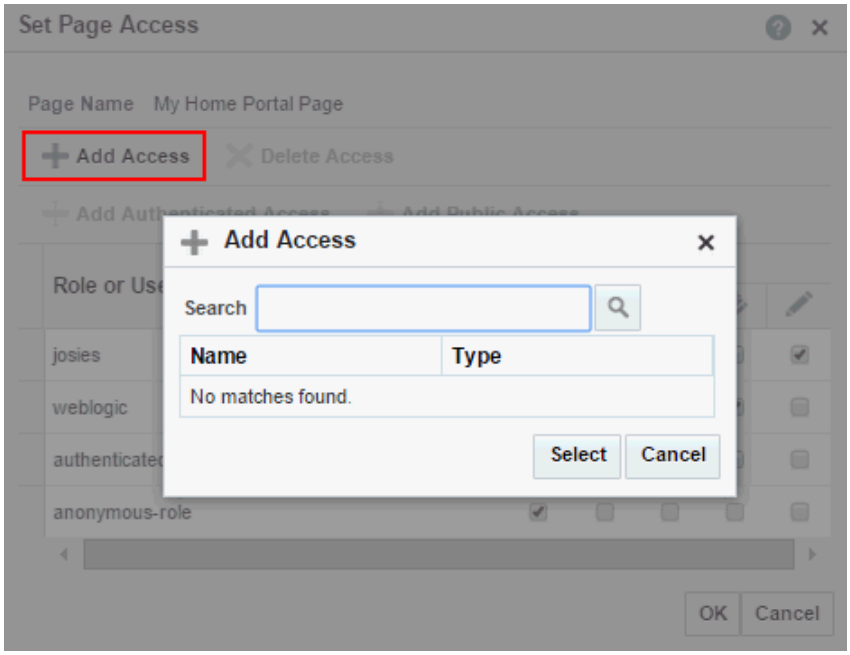
   The role `anonymous-role` is added under **Role or User** with default **View** access to the page.

Figure 41-6 Anonymous Role Access



5. To grant page access permissions to selected users and roles, click **Add Access** to open the Add Access dialog.

Figure 41-7 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 Searching for a User or Group in the Identity Store in *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:

Table 41-2 Page Access Privileges in the Set Page Access Dialog

Page Access	Role or User Permissions
View Page	Access the page for viewing, but cannot perform any other actions on the page. Other permissions do not implicitly include this privilege
Edit Page	Edit the page using the page editor. This includes adding, rearranging, and deleting content; renaming the page; and changing page properties. This permission additionally requires the **View Page** permission.
Delete Page	Delete the page. This permission additionally requires the **View Page** permission.
Perform All Page Actions	Perform all actions on the page.
Personalize Page	Adjust a user's own view of a page. This includes rearranging page content, collapsing and restoring page content, and removing page content. This permission additionally requires the **View Page** permission.

> **Tip:**
>
> By default, all authenticated users and user roles that you add are granted page view access. The other access privileges must be explicitly granted.

8. To revoke access to the page, select the role or user, and click **Delete Access**.
9. Click **OK**.

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.

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 Setting Access on a Built-in Business Role Page.

To make a custom business role page public:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Business Role Pages**.

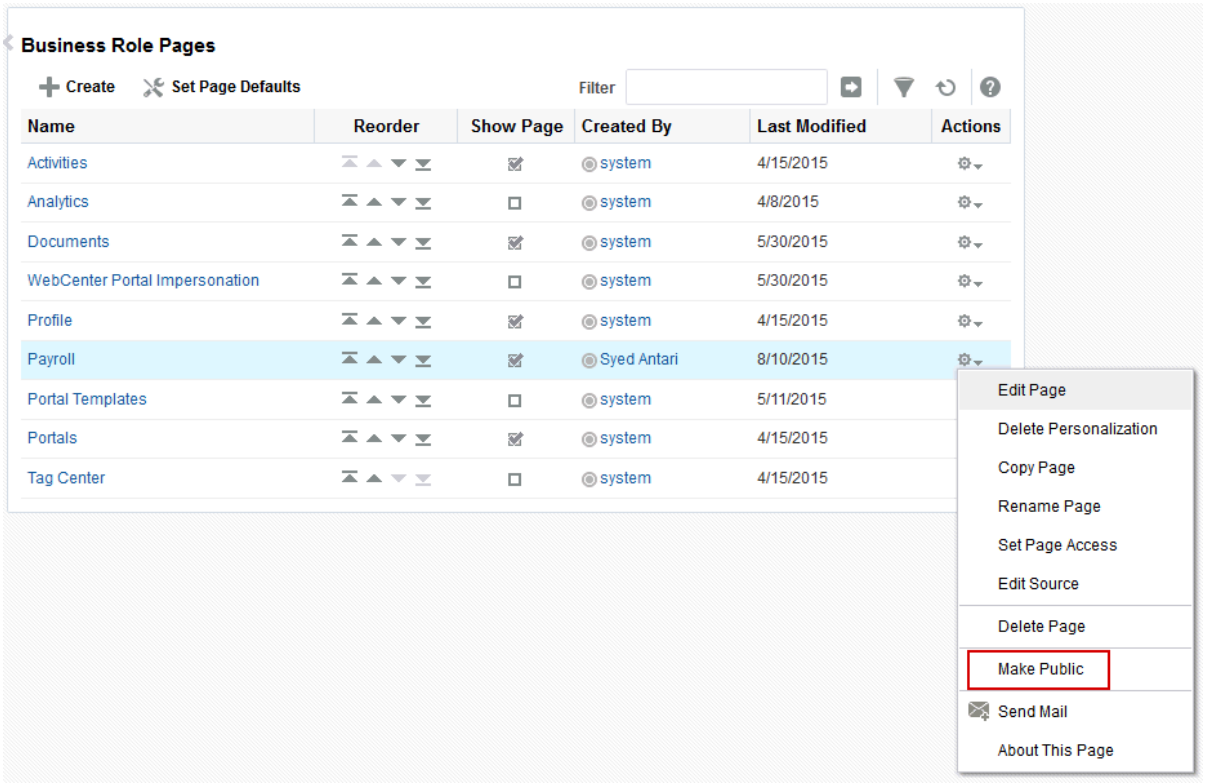
   You can also enter the following URL in your browser to navigate directly to the **Business Role Pages** page:

   `http://host:port/webcenter/portal/admin/settings/businessrolepages`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *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 41-8).

Figure 41-8 Make Public Option on a Custom Business Role Page



Setting Access on a Built-in Business Role Page

Built-in business role pages, such as Activities and Portals, are available to all users by default (see Table 41-1). You cannot modify the security of built-in business role pages through WebCenter Portal. If you want to change the default security settings, for example, you want to hide a built-in 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 built-in 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_Portal',toLocation='/scratch/
   mdsdump', docs='/oracle/webcenter/page/scopedMD/
   s8bba98ff_4cbb_40b8_beee_296c916a23ed/role/anonymous-role/pages.xml')

   exportMetadata(application='webcenter',server='WC_Portal',toLocation='/scratch/
   mdsdump', docs='/oracle/webcenter/page/scopedMD/
   s8bba98ff_4cbb_40b8_beee_296c916a23ed/role/authenticated-role/pages.xml')
   ```

   Where `toLocation` specifies a target directory on your system for the file you want to export. For detailed syntax, see `exportMetadata` in *WLST Command Reference for WebLogic Server*.

For information on how to run WLST commands, see 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_Portal',fromLocation='/
scratch/mdsdump', docs='/oracle/webcenter/page/scopedMD/
s8bba98ff_4cbb_40b8_beee_296c916a23ed/role/anonymous-role/pages.xml')
importMetadata(application='webcenter',server='WC_Portal',fromLocation='/
scratch/mdsdump', docs='/oracle/webcenter/page/scopedMD/
s8bba98ff_4cbb_40b8_beee_296c916a23ed/role/authenticated-role/pages.xml')
```

   Where `fromLocation` specifies the directory that contains the file you want to import. For detailed syntax, see `importMetadata` in *WLST Command Reference for WebLogic Server*.

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

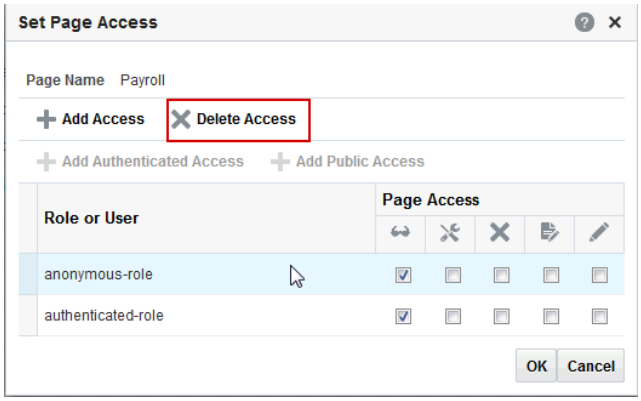
> **Note:**
>
> By default, any authenticated or anonymous user role will not be able to view the Activity Stream page (used as an example here). However, if the user logs into WebCenter Portal, from the **Personalize Pages** page the user can override this setting and make the page visible using the **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`.

Revoking Access to a Custom Business Role Page

To revoke access privileges to a custom business role page:

1. Follow the steps in 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 41-9).

Figure 41-9 Delete Access Option in Set Page Access Dialog



3. Click **Delete** in the confirmation dialog.

Showing and Hiding Business Role Pages

To show or hide a business role page in Home portal navigation for all authorized users:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Business Role Pages**.

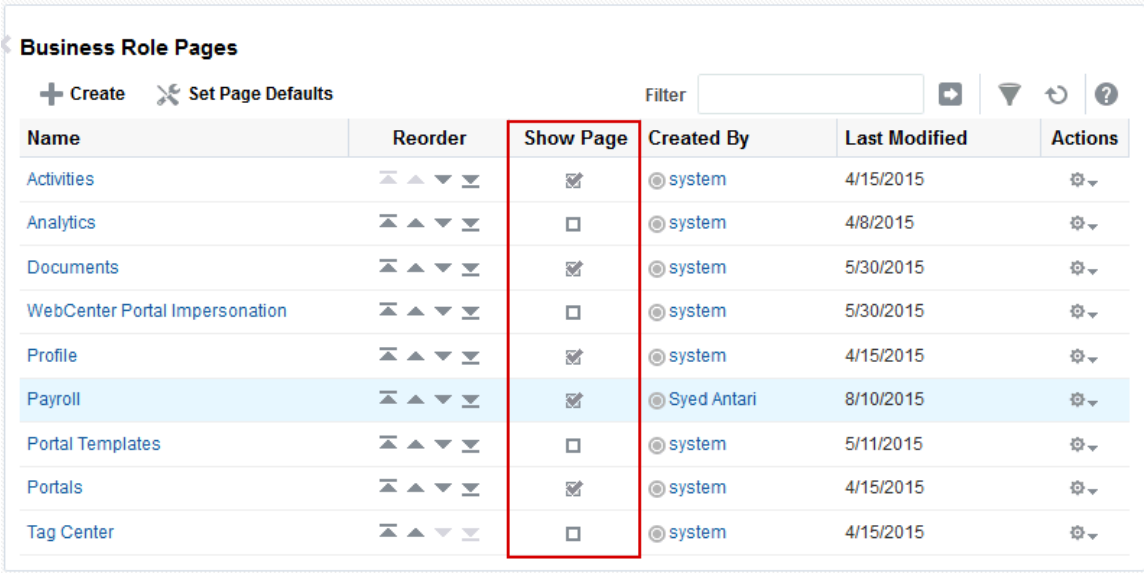
   You can also enter the following URL in your browser to navigate directly to the **Business Role Pages** page:

```
http://host:port/webcenter/portal/admin/settings/businessrolepages
```

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. For the page you want to show or hide (Figure 41-10):
   - 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 41-10 Show Page Option for Business Role Pages



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 Rearranging Page Order in the Home Portal in *Using Portals in 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 **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Business Role Pages**.

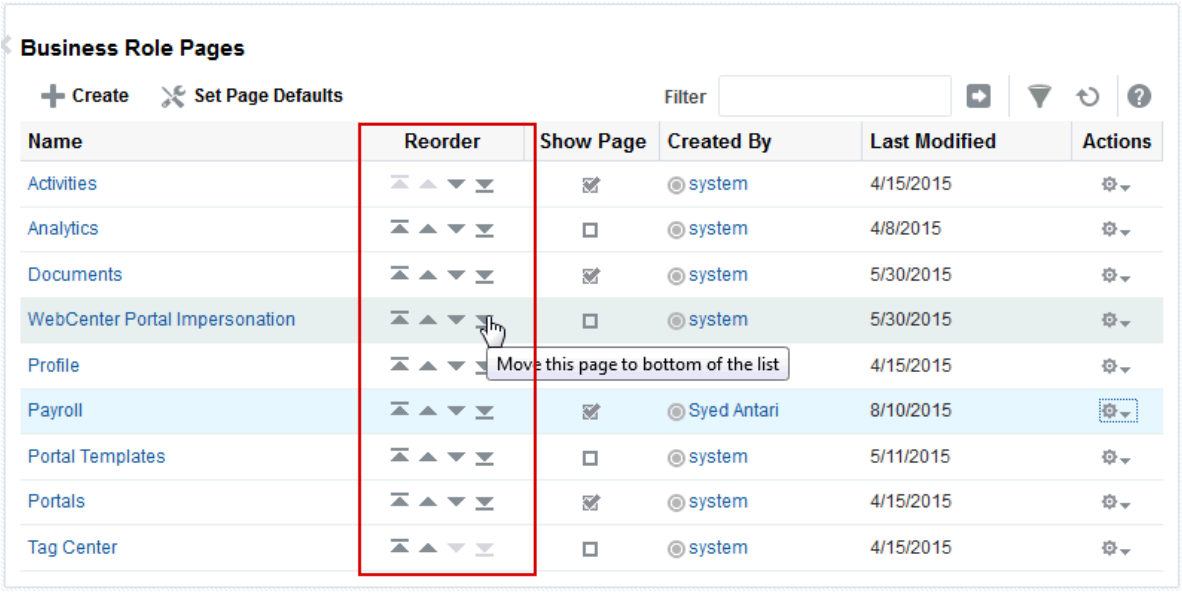
   You can also enter the following URL in your browser to navigate directly to the **Business Role Pages** page:

   `http://host:port/webcenter/portal/admin/settings/businessrolepages`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *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 41-11).

Figure 41-11 Reorder Icons on Business Role Pages



Alternatively, drag and drop pages into the desired order.

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 Editing a Page in *Building Portals with Oracle WebCenter Portal* and Editing a Personal Page in *Using Portals in Oracle WebCenter Portal*.

As the WebCenter Portal system administrator, you can also initiate an edit of a business role page.

To edit a custom business role page:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Business Role Pages**.

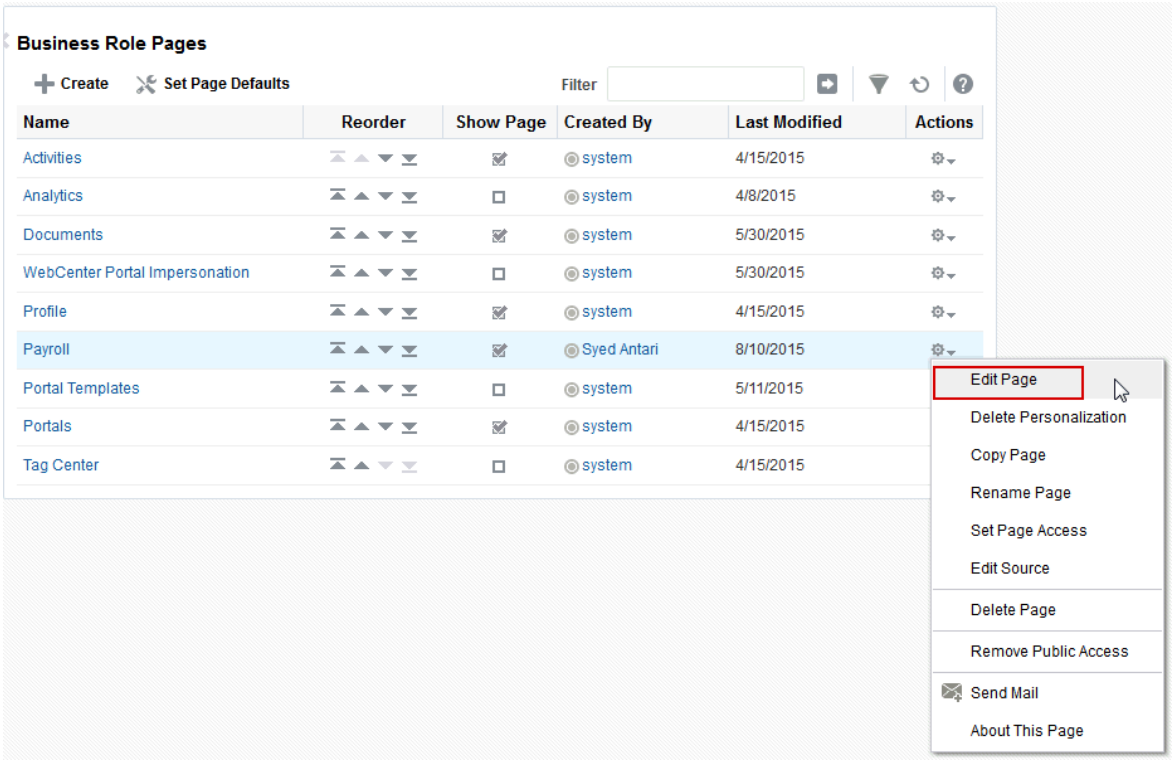
   You can also enter the following URL in your browser to navigate directly to the **Business Role Pages** page:

   `http://host:port/webcenter/portal/admin/settings/businessrolepages`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *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 41-12).

Figure 41-12 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 Editing a Page in *Building Portals with Oracle WebCenter Portal*.

3. Edit the page, and click **Save** and then **Close** when you have finished.

Editing the Source of a Business Role Page

If you have the `Edit Page` permission on a Business Role page, you can edit the source of the page without opening the page in Composer.

To edit the source of a custom Business Role page:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Business Role Pages**.

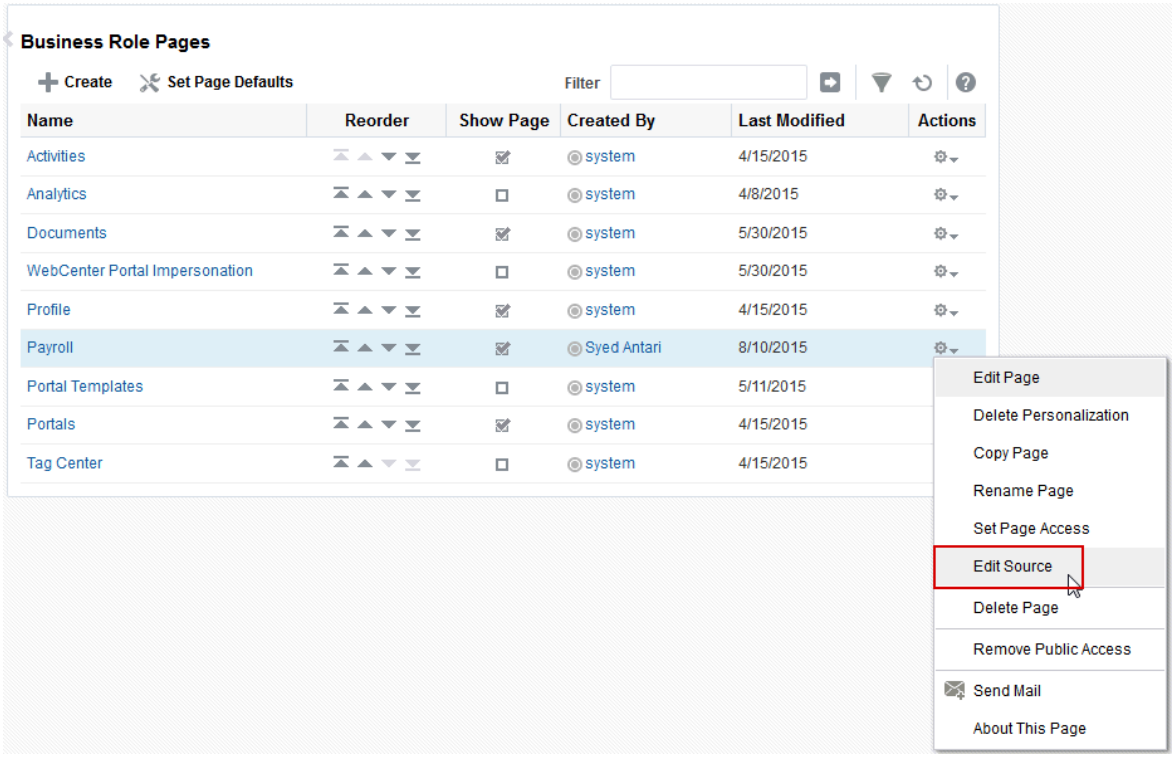
   You can also enter the following URL in your browser to navigate directly to the **Business Role Pages** page:

   `http://host:port/webcenter/portal/admin/settings/businessrolepages`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

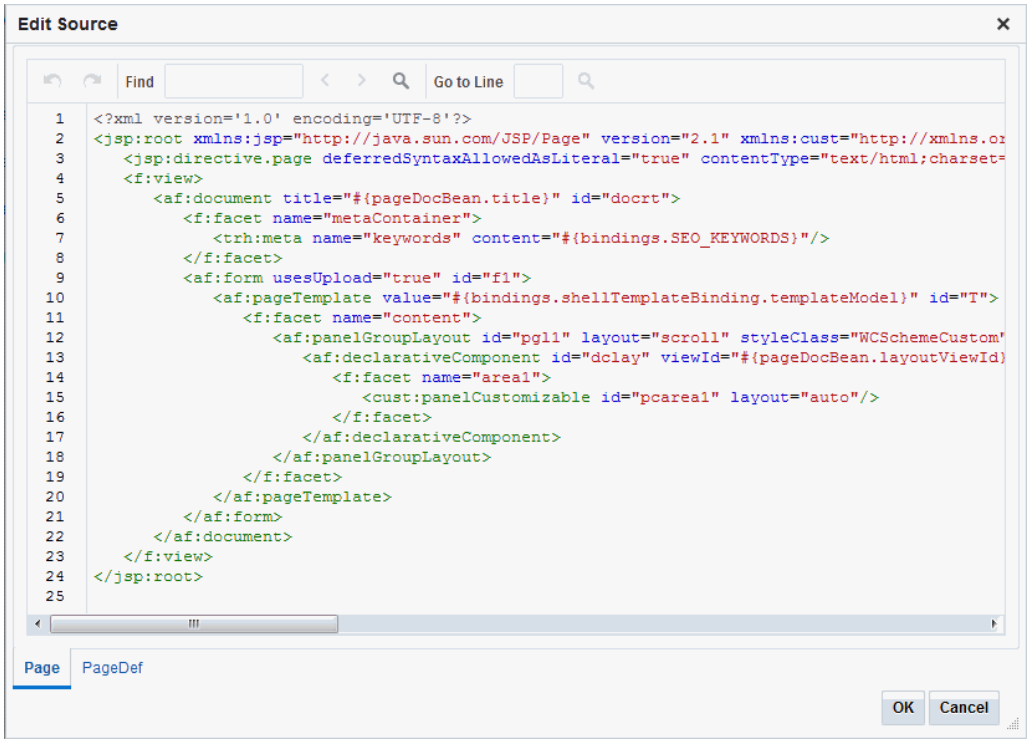
2. Click the **Actions** icon for the custom page whose source you want to edit, and select **Edit Source** (Figure 41-13).

Figure 41-13 Edit Source Option on a Custom Business Role Page



The Edit Source dialog opens (Figure 41-14).

Figure 41-14 Edit Source Dialog



3. Edit the page source, as desired.

   For more information about editing the source of a page, see Viewing and Modifying Page Source Code in *Building Portals with Oracle WebCenter Portal*.

4. Click **OK**.

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 Specifying the Target Audience for a Business Role Page). You cannot copy custom business role pages.

To copy a built-in business role page:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Business Role Pages**.

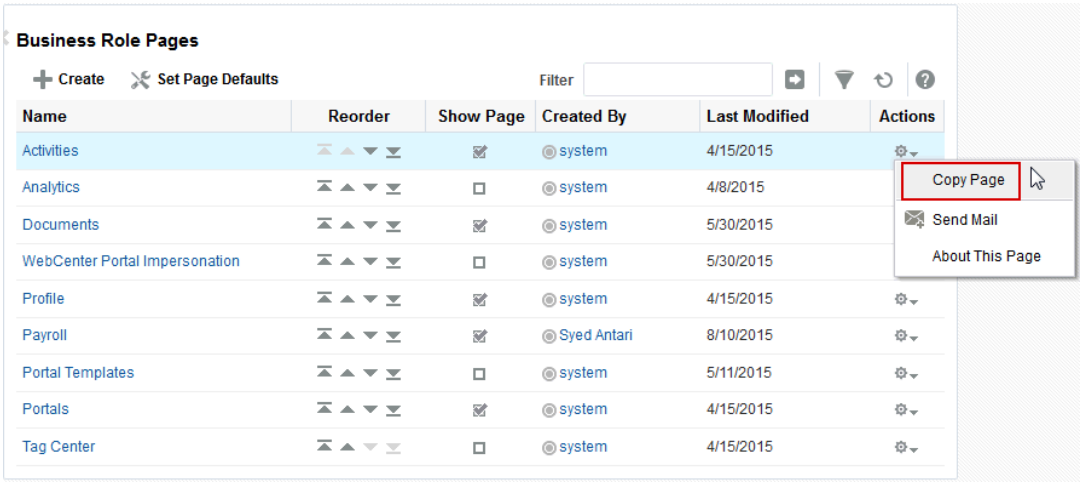
   You can also enter the following URL in your browser to navigate directly to the **Business Role Pages** page:

   `http://host:port/webcenter/portal/admin/settings/businessrolepages`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

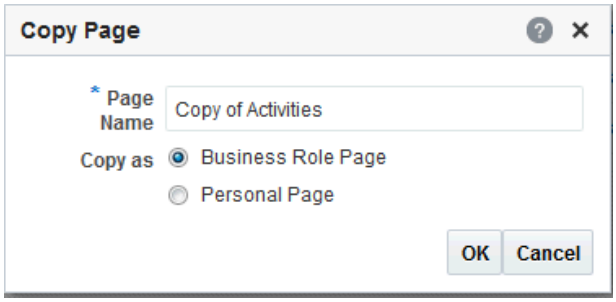
2. Select the page, click the **Actions** icon for the page you want to copy, and select **Copy Page** (Figure 41-15).

Figure 41-15 Copy Page Option on a Built-in Business Role Page



3. In the Copy Page dialog, enter a name for the new page (Figure 41-16).

Figure 41-16 Copy Page Dialog



4. Next to **Copy as**, specify whether the page will be copied as 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 Editing a Page in *Building Portals with Oracle WebCenter Portal*.

6. Optionally, edit the page, and click **Save** when you have finished.

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 custom business role page:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), 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/admin/settings/businessrolepages`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *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.

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 Deleting a Page in *Building Portals with Oracle WebCenter Portal*. As the WebCenter Portal system administrator, you can delete custom business role pages.

> **Note:**
>
> Built-in 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 **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Business Role Pages**.

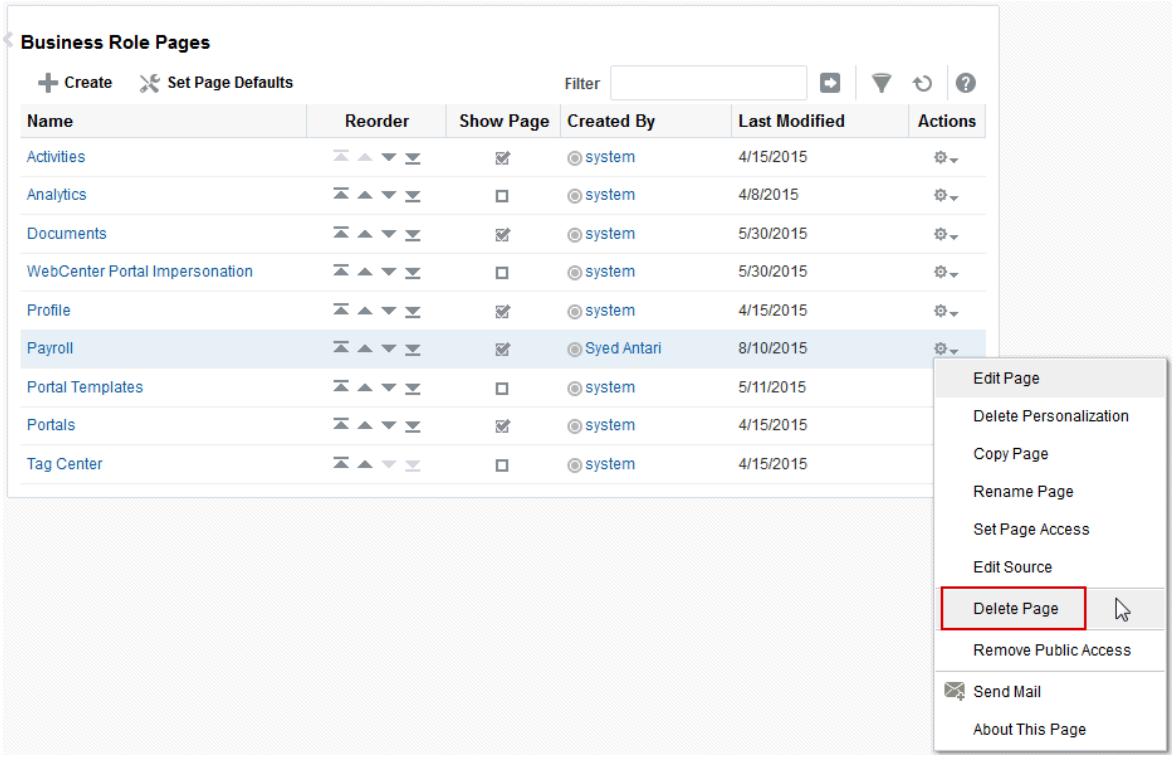
   You can also enter the following URL in your browser to navigate directly to the **Business Role Pages** page:

   `http://host:port/webcenter/portal/admin/settings/businessrolepages`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click the **Actions** icon for the page you want to delete, and select **Delete Page** (Figure 41-17).

Figure 41-17 Delete Page Option on a Custom Business Role Page



3. In the confirmation dialog, click **Delete**.

42

Managing Personal Pages

Use the **Personal Pages** page in WebCenter Portal Administration to administer personal pages of all users 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.

> **Permissions:**
>
> To perform the tasks in this chapter, you must have the WebCenter Portal `Administrator` role or a custom role that grants the following permissions:
>
> - `Portal Server: Manage All` or `Portal Server: Manage Configuration`
> - `Pages: Create, Edit, and Delete Pages`
>
> For more information about permissions, see About Application Roles and Permissions.

Topics:

- About Personal Page Administration
- Setting Application-Level Page Creation Defaults for Personal Pages
- Preventing Users from Creating Personal Pages
- Providing Navigation to Personal Pages
- Changing Access Permissions on a Personal Page
- Editing a Personal Page
- Editing the Source of a Personal Page
- Copying a Personal Page
- Removing All User Customizations from a Personal Page
- Deleting a Personal Page

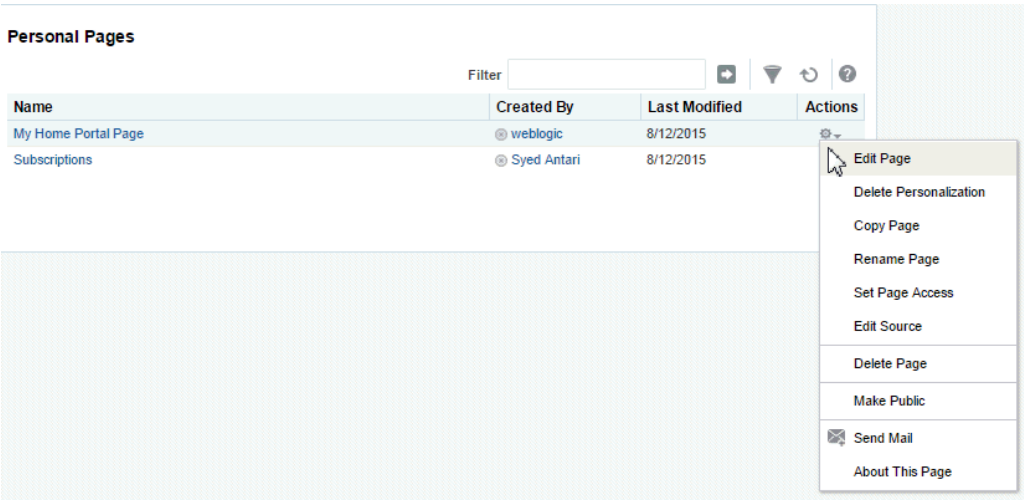
About Personal Page Administration

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 the **Personal Pages** page in WebCenter Portal Administration. An **Actions** menu is associated with each listed

page, providing access to options for editing in the page editor, removing user customizations, copying, renaming, securing, editing the source, deleting, and making the personal page public (Figure 42-1).

Figure 42-1 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.

Setting Application-Level Page Creation Defaults for Personal Pages

In addition to the page creation defaults authorized users can set for themselves (see Setting Page Creation Defaults for Personal Pages in *Using Portals in 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 application—level configuration of personal pages is done on the **Business Role Pages** page in WebCenter Portal Administration. For more information, see 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.

Preventing Users from Creating Personal Pages

The application-level `Pages: Create Pages` permission allows users to create personal pages in the Home portal. You can revoke this permission from individual users to prevent them from creating personal pages. For more information, see

To assign permissions to users, you assign them a role than includes the permissions they need. To assign a user a role that includes or excludes the `Pages: Create Pages` permission, see Assigning Users (and Groups) to Application Roles.

Providing Navigation to Personal Pages

If you want to add a link to a personal page in a portal's navigation, see Creating and Managing Personal Pages in *Using Portals in Oracle WebCenter Portal*. For detailed information about working with portal navigation, see Working with Navigation Task Flow Propertiesin *Building Portals with Oracle WebCenter Portal*.

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 **Settings** tab, click **Personal Pages**.

   You can also enter the following URL in your browser to navigate directly to the **Personal Pages** page:

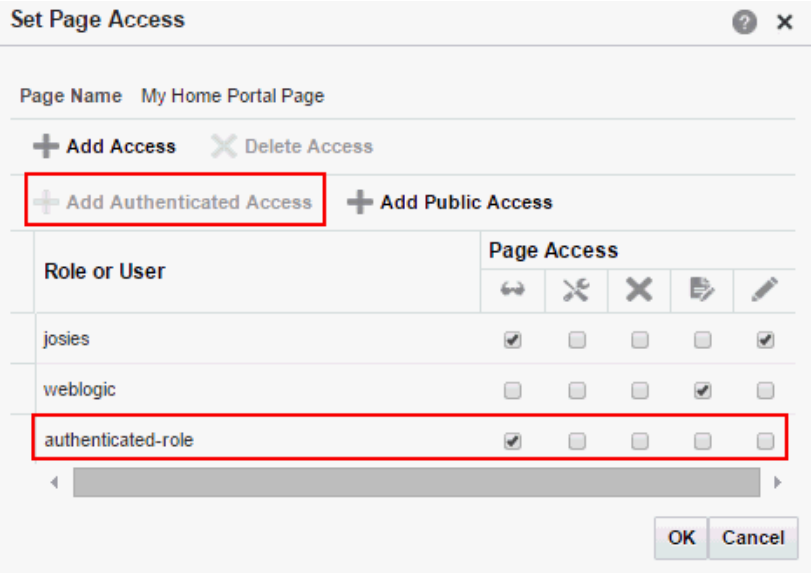
   `http://host:port/webcenter/portal/admin/settings/personalpages`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click the **Actions** icon for the page you want to secure, and select **Set Page Access** to open the Set Page Access dialog.
3. To grant page access permissions to all authenticated users (that is, to users who are logged in to WebCenter Portal), click **Add Authenticated Access**.

   The role `authenticated-role` is added under **Role or User** with default **View** access to the page.

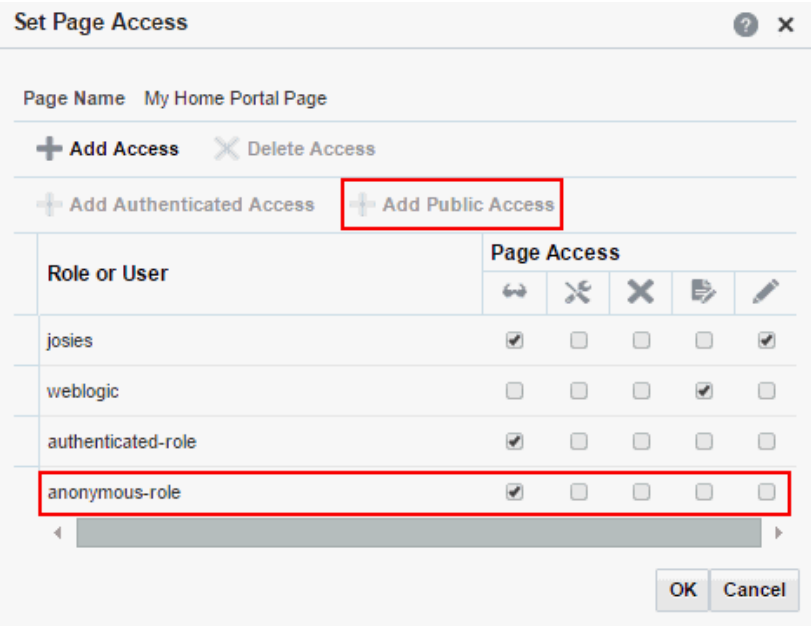
Figure 42-2 Authenticated Role Access



4. To grant page access permissions to all public users (that is, users who have not logged in to WebCenter Portal as well as those who have) click **Add Public Access**.

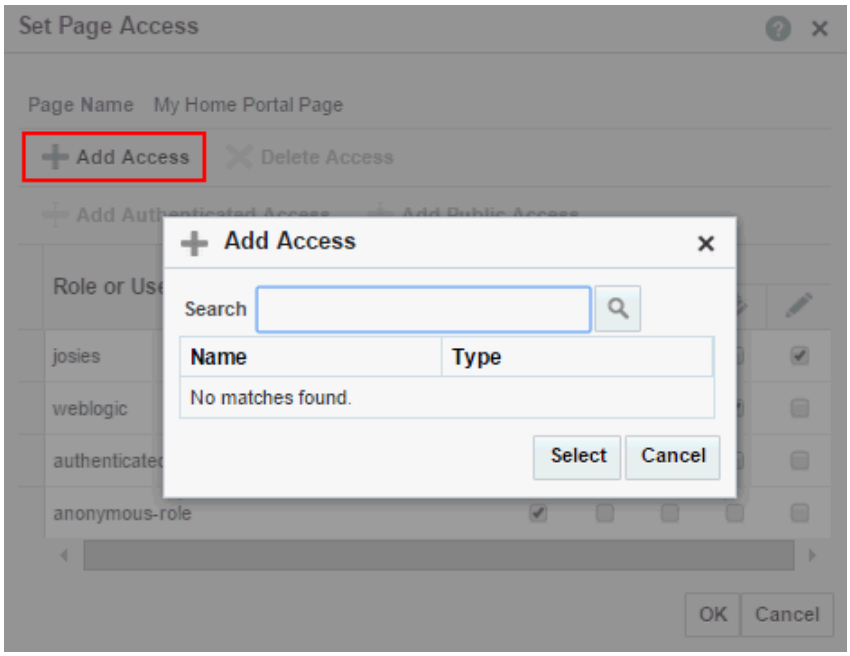
   The role `anonymous-role` is added under **Role or User** with default **View** access to the page.

Figure 42-3 Anonymous Role Access



5. To grant page access permissions to selected users and roles, click **Add Access** to open the Add Access dialog.

Figure 42-4 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 Searching for a User or Group in the Identity Store in *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:

Table 42-1 Page Access Privileges in the Set Page Access Dialog

Page Access	Role or User Permissions
View Page	Access the page for viewing, but cannot perform any other actions on the page. Other permissions do not implicitly include this privilege
Edit Page	Edit the page using the page editor. This includes adding, rearranging, and deleting content; renaming the page; and changing page properties. This permission additionally requires the **View Page** permission.
Delete Page	Delete the page. This permission additionally requires the **View Page** permission.
Perform All Page Actions	Perform all actions on the page.
Personalize Page	Adjust a user's own view of a page. This includes rearranging page content, collapsing and restoring page content, and removing page content. This permission additionally requires the **View Page** permission.

> **Tip:**
>
> By default, all authenticated users and user roles that you add are granted page view access. The other access privileges must be explicitly granted.

8. To revoke access to the page, select the role or user, and click **Delete Access**.
9. Click **OK**.

Editing a Personal Page

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 Editing the Source of a Personal Page.

To edit a personal page:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), 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/admin/settings/personalpages`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click the **Actions** icon for the page you want to edit, and select **Edit Page** (see Figure 42-1) to open the page in the page editor.

> **See Also:**
>
> For information about editing a page, see Editing a Page in *Building Portals with Oracle WebCenter Portal*.

3. Update the page, and click **Save** and then **Close** when you have finished.

Editing the Source of a Personal Page

You can edit the source of a personal page without opening the page in the page editor.

To edit the source of a personal page:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Personal Pages**.

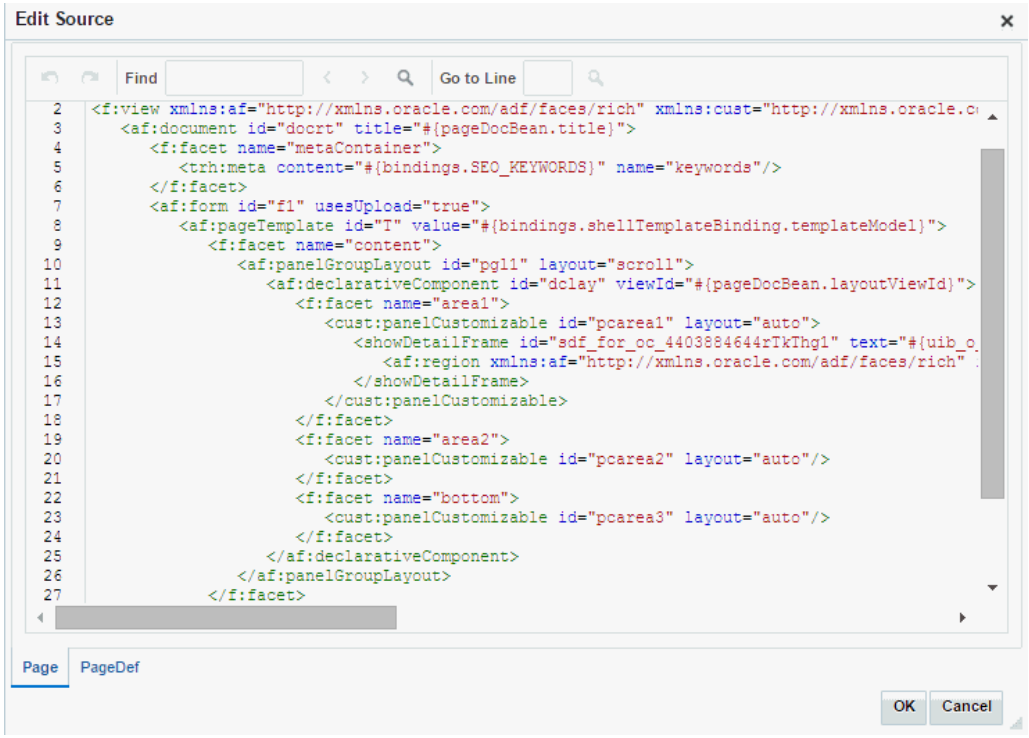
   You can also enter the following URL in your browser to navigate directly to the **Personal Pages** page:

   `http://host:port/webcenter/portal/admin/settings/personalpages`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click the **Actions** icon for the page whose source you want to edit, and select **Edit Source** (Figure 42-1) to open the Edit Source dialog (Figure 42-6).

Figure 42-5 Edit Source Dialog



3. Edit the page source, as desired.
4. Click **OK**.

Copying a Personal Page

As the system administrator, you are authorized to copy any page in the Oracle 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 Specifying the Target Audience for a Business Role Page.

To copy a personal page:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Personal Pages**.

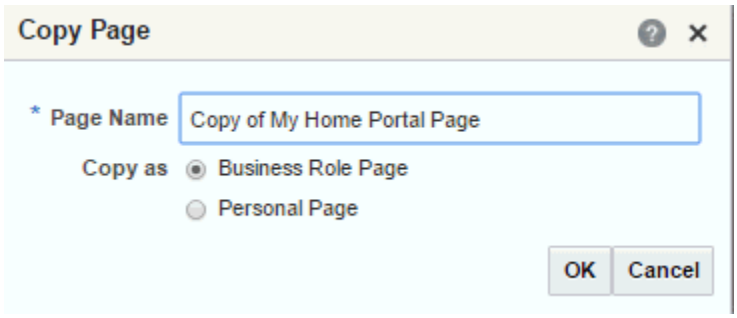
   You can also enter the following URL in your browser to navigate directly to the **Personal Pages** page:

   `http://host:port/webcenter/portal/admin/settings/personalpages`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click the **Actions** icon for the page you want to copy, and select **Copy Page** (Figure 42-1) to open the Copy Page dialog (Figure 42-6).

Figure 42-6 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 learn more about copying a page, see Copying a Page in *Building Portals with Oracle WebCenter Portal*.
5. Click **OK**.

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 **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), 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/admin/settings/personalpages`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click the **Actions** icon for the target page, and select **Delete Personalization** (Figure 42-1).
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 sizes, collapsed task flows are expanded, and so on.

Deleting a Personal 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 **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), 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/admin/settings/personalpages`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click the **Actions** icon for the page you want to delete, and select **Delete Page** (Figure 42-1).
3. In the confirmation dialog, click **Delete**.

43

Administering Device Settings

Use the **Device Settings** page in WebCenter Portal Administration to control how portals render on different kinds of devices including desktop browsers, smart phones, and tablets.

> **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 All` or `Portal Server: Manage Configuration`
>
> For more information about permissions, see About Application Roles and Permissions.

Topics:

- About Device Settings
- Creating and Managing Devices
- Creating and Managing Device Groups
- Enabling Page Variants for Device Groups
- Managing Device and Device Group Lifecycles
- Previewing Devices
- Guidelines and Best Practices for Device Settings
- Discovering Device Attributes: A Sample Task Flow

About Device Settings

To successfully manage and administer device settings, you need to be familiar with the concepts described in this section.

- Introduction to Device Settings
- What Are Devices?
- What Are Device Groups?
- Other Related Concepts
- Basic Use Case: Adding Support for a New Device
- Understanding How Device Settings are Applied

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.

What Are Devices?

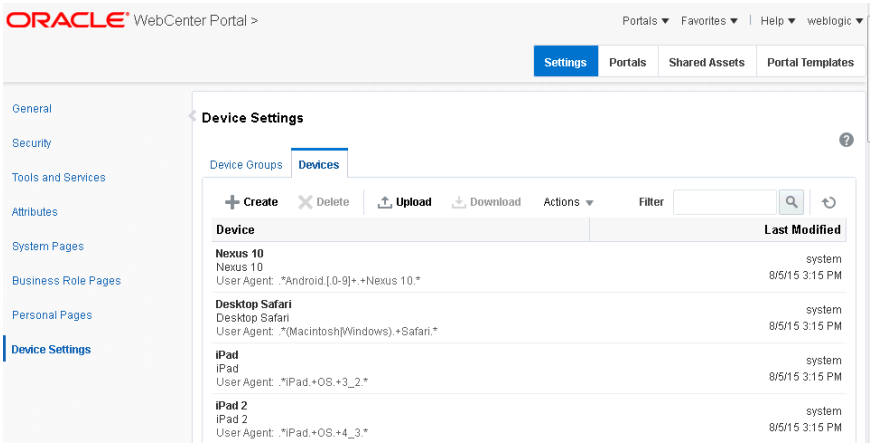
A *device* is a representation in WebCenter Portal of a physical device, like a smart phone or tablet, that users use to interact with a portal. Each time a portal page is requested, WebCenter Portal determines the type of device from which the request originated. This information enables the portal to decide what category of devices or "device group" the device is associated with.

See 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 43-1 shows some of the default devices listed in WebCenter Portal Administration.

Figure 43-1 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 WebCenter Portal 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 Creating and Managing Devices .

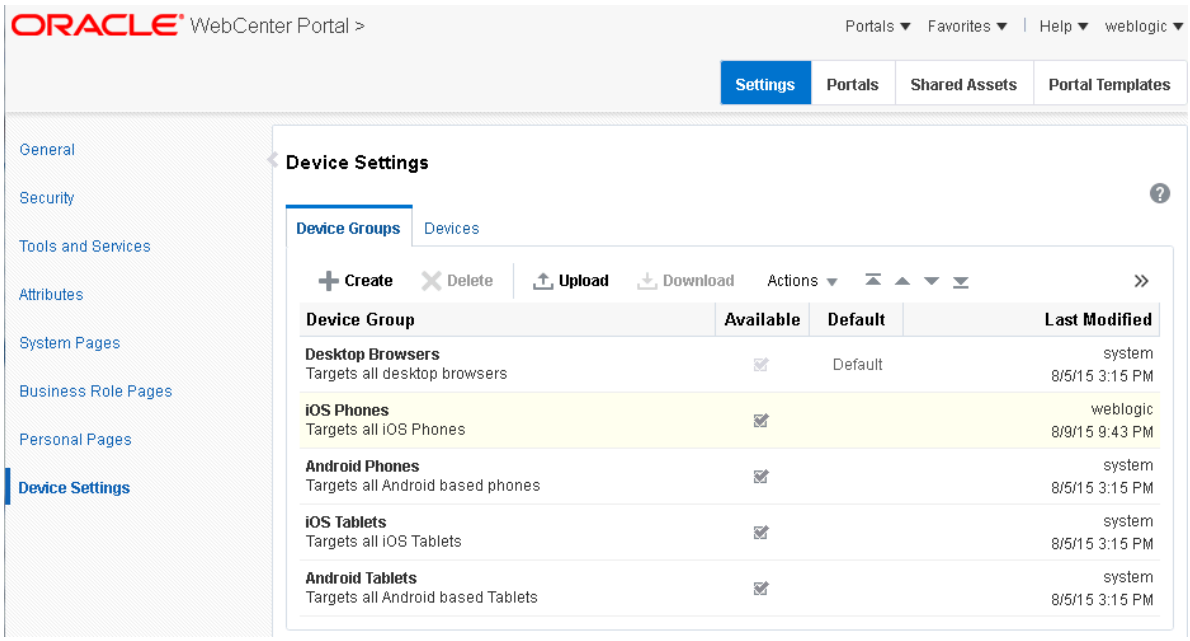
What Are Device Groups?

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 43-2 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 Creating and Managing Device Groups.

Figure 43-2 WebCenter Portal Administration: Device Settings Page



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 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.

  For portals that do not use responsive page templates, you can define page variants that optimize the display of a portal page for a specific device group. At runtime, WebCenter Portal can check if a page variant exists for the device being used to display a portal page, and displays the page using the page variant definition. By default, this check is not performed. To turn it on, see Enabling Page Variants for Device Groups.

  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 Creating a Page Variant of a System Page for Device Groups. For information on creating page variants for portal pages, see Creating a Page Variant for a Device Group in *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 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 Setting Page Behavior for a Specific Page When No Page Variant Exists in *Building Portals with Oracle WebCenter Portal*.

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

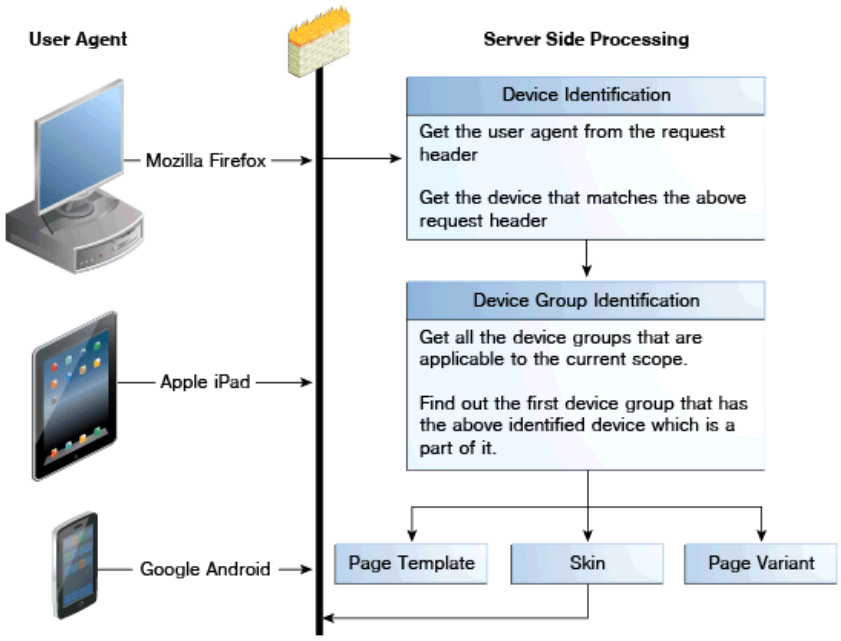
does not render properly on this device–there is a lot of white space and the company logo doesn't look right.You are asked to support this new device. The basic steps are:

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 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 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.

Understanding How Device Settings are Applied

Figure 43-3 illustrates the flow of how WebCenter Portal handles requests from multiple different devices.

Figure 43-3 How the Portal Handles Requests from Different Devices



As Figure 43-3 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 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 and page variants are enabled) are returned and the page renders on the device. If a page does not have a page variant, or if page variants are disabled, 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 Setting the Page Behavior for a Portal When No Page Variant Exists in *Building Portals with Oracle WebCenter Portal*.

Creating and Managing Devices

This section explains how to create and manage devices:

- Creating a New Device
- Editing a Device
- Copying a Device
- Filtering the List of Devices
- Deleting a Device

> **Note:**
>
> In some cases, you may be unable to view page variants after a device configuration is added or modified on the **Device Settings** page. In this case, log out and log in again to clear the cache, and after that, the device will be recognized correctly. Additionally, make sure that page variants are enabled (seeEnabling Page Variants for Device Groups).

Creating a New Device

To create a new device:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), 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/admin/settings/device`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. On the **Devices** tab, click **Create**.

   The Create Device page displays, containing three sections: **Device**, **Optional Attributes**, and **Additional Attributes**.

3. 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 WebCenter Portal 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.

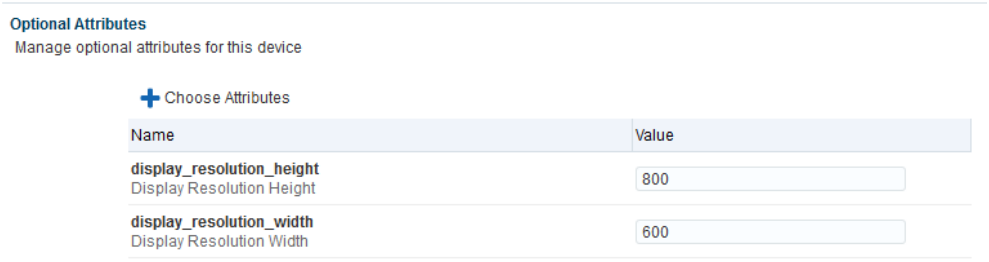
     > **Note:**
     >
     > The user agent string is a regular expression and conforms to the syntax specified by the Java platform (java.util.regex.Pattern). As such, certain special characters might need to be escaped with `\` if you want to match them. These characters include `[\^$.|?*+()` and, in some cases, curly braces `{}`. For example, a parenthesis must be escaped as follows: `\(`. For further guidance, refer to a good reference on regular expression syntax.

   - **Description** - (Optional) Specify a description that helps to identify the purpose of the device.

4. Use the **Optional Attributes** section to manage attributes such as display resolution height and width. You can edit their default values as required.

   > **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 43-4 Specifying Optional Attributes



5. (Optional) 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.

6. Click **Create**.

Editing a Device

To edit an existing device:

> **Note:**
>
> In some cases, you may be unable to view page variants after a device configuration is added or modified on the **Device Settings** page. In this case, log out and log in again to clear the cache, and after that, the device will be recognized correctly. Additionally, make sure that page variants are enabled (see Enabling Page Variants for Device Groups).

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), 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/admin/settings/device`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. On the **Devices** tab, select the device you wish to edit, then click the **Actions** menu and select **Edit**.
3. Edit the device settings. For information about the device settings that can be edited, see Creating a New Device .
4. Click **Save**.

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 **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), 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/admin/settings/device`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. On the **Devices** tab, select the device you wish to copy, then click the **Actions** menu and select **Copy**.
3. In the Copy dialog, specify the name, display name, user agent, and description of the device.
4. Click **OK**.

   The copied device appears in the **Devices** list.

Filtering the List of Devices

The **Filter** field lets you filter the list of devices shown in the **Devices** table. Filtering searches on device group names, display names, descriptions, and user agents.

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 **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), 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/admin/settings/device`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. On the **Devices** tab, select the device you wish to delete. Press 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 provided with WebCenter Portal.

3. Click **Delete**.
4. Confirm your action in the Delete Device dialog.

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.

- Creating a Device Group
- Editing a Device Group
- Copying a Device Group
- Showing and Hiding Device Groups
- Setting a Default Device Group
- Ordering Device Groups
- Filtering Device Groups
- Deleting a Device Group

See also Basic Use Case: Adding Support for a New Device.

> **Note:**
>
> In some cases, you may be unable to view page variants after a device configuration is added or modified on the **Device Settings** page. In this case, log out and log in again to clear the cache, and after that, the device will be recognized correctly. Additionally, make sure that page variants are enabled (see Enabling Page Variants for Device Groups).

Creating a Device Group

To create a device group:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **Device Settings**.

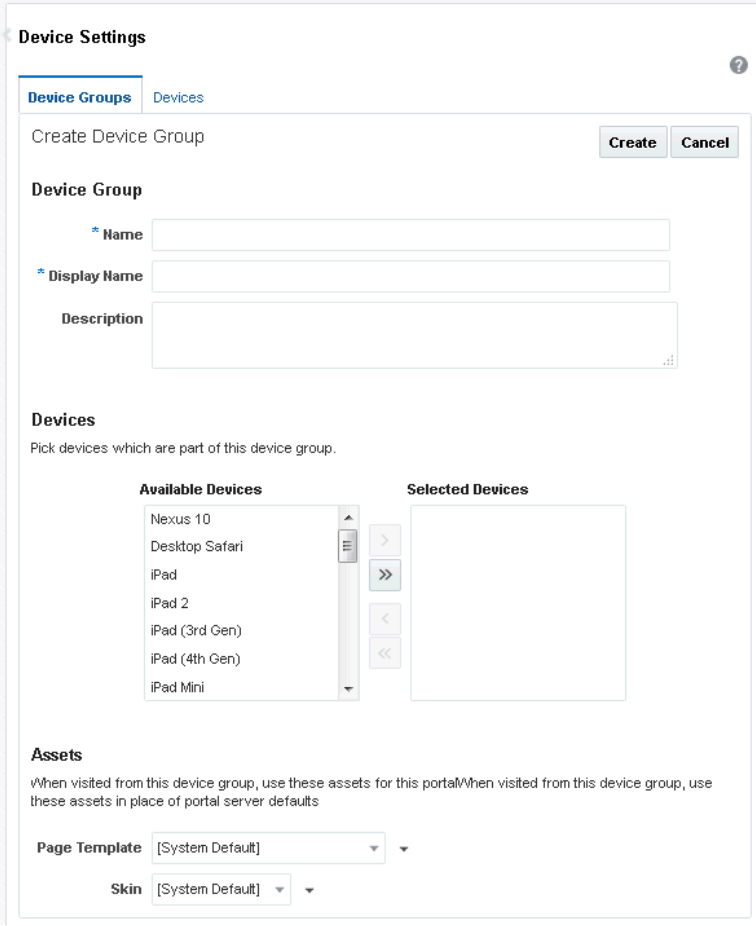
   You can also enter the following URL in your browser to navigate directly to the **Device Settings** page:

   `http://host:port/webcenter/portal/admin/settings/device`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. On the **Device Groups** tab, click **Create**.

Figure 43-5 Creating a Device Group



3. On the Create Device Group page, 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 WebCenter Portal. 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 Expression Language Expressions in *Developing for Oracle WebCenter Portal*.

6. Click **Create**.

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 **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), 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/admin/settings/device`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. On the **Device Groups** tab, 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 Expression Language Expressions in *Developing for Oracle WebCenter Portal*.

6. Click **Save**.
7. Click **Close** to close the Edit Device Group page.

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 **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), 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/admin/settings/device`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. On the **Device Groups** tab, select the device group you wish to edit, 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 43-6).

Figure 43-6 Copying a Device Group

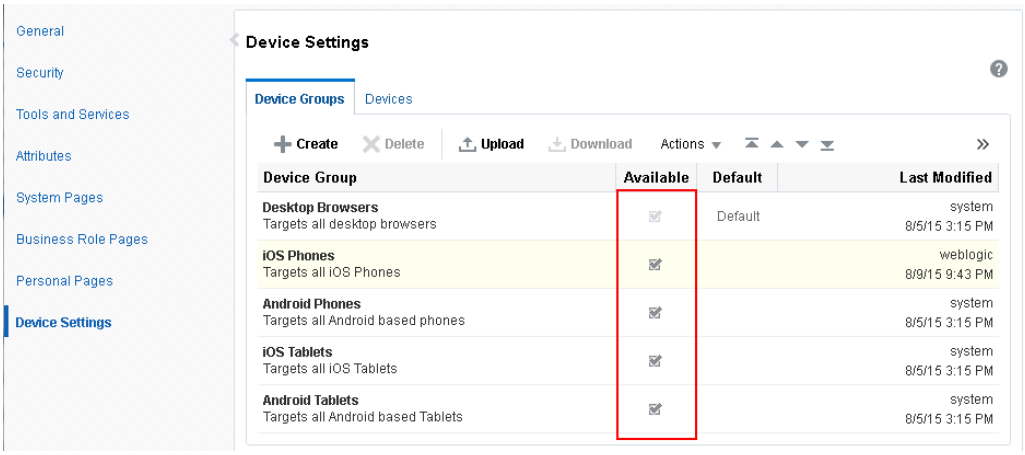
4. Click **OK**.

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 43-7). 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 manager. See also Creating a Page Variant for a Device Group in *Building Portals with Oracle WebCenter Portal*.

When you create a device group, by default, it is marked as unavailable.

Figure 43-7 Available and Hidden Device Groups



To show or hide a device group:

1. On the **Settings** 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/admin/settings/device`

2. On the **Device Groups** tab, in the **Available** column, select or deselect the check box in the **Available** column to show or hide the device group.

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 **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), 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/admin/settings/device`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. 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.

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 **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), 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/admin/settings/device`

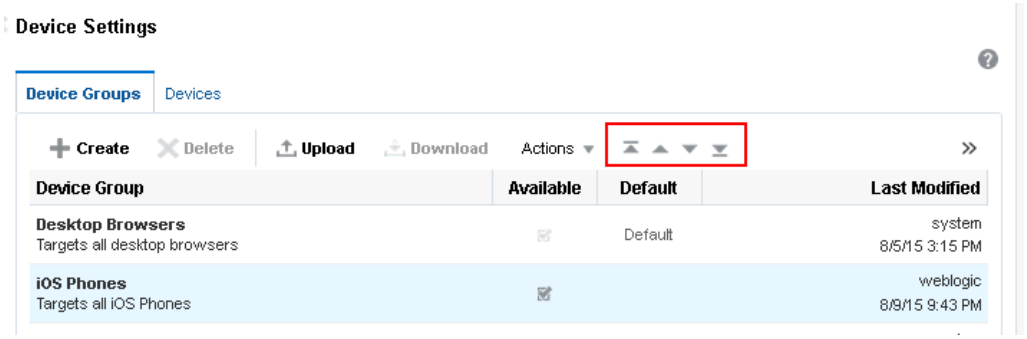
   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. On the **Device Groups** tab, 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 43-8 Reordering Device Groups



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.

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 Guidelines and Best Practices for Device Settings.

To delete a device group:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), 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/admin/settings/device`

> **See Also:**
>
> WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. On the **Device Groups** tab, select the device group that you want to delete, then click **Delete**.
3. In the Delete Device Group dialog, click **Delete**.

Enabling Page Variants for Device Groups

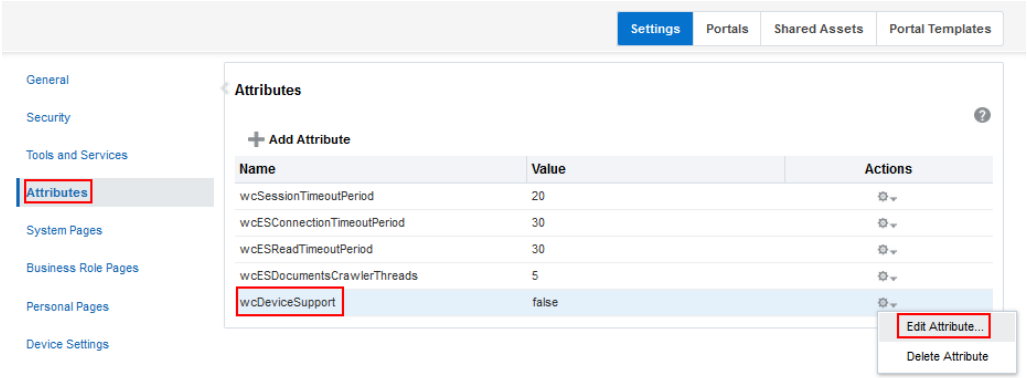
By default, WebCenter Portal does not check for page variants at runtime. For portals that have page variants defined for different device groups, you must turn on the runtime check for page variants by setting the `wcDeviceSupport` attribute to `true`. By default, this attribute is set to `false`.

> **Note:**
>
> Portal templates and skins that are specific to a device group are honored by default. Only the check for page variants is disabled.

To enable page variants at runtime:

1. On the **Settings** page, click **Attributes**.
2. On the **Attributes** page, click the **Actions** icon for the `wcDeviceSupport` attribute and select **Edit Attribute**.

Figure 43-9 Attributes Page



3. In the Edit Attribute dialog, set the **Value** to `true`.
4. Click **OK**.
5. Restart WebCenter Portal.

Managing Device and Device Group Lifecycles

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 lifecycle mechanism described in the following sections:

- Downloading a Device Group or Device
- 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 built-in 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.

Downloading a Device Group or Device

To download a device group or a device to a file:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), 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/admin/settings/device`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. On the **Device Groups** or **Devices** tab, select the device group or device you wish to download. Press Ctrl+click to select multiple rows.
3. Click **Download**.
4. In the Download dialog, in the **Archive File Name** field, enter a name for the device group or the device archive file. The archive file must have the `.aar` extension.
5. 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. The `.aar` archive file is saved to the default path *`DOMAIN_HOME`*`/WC_Archives`, where *`DOMAIN_HOME`* refers to the domain location where WebCenter Portal is installed.

  See also Downloading an Asset in *Building Portals with Oracle WebCenter Portal*.

6. Click **Download**.

Uploading a Device Group or Device

To upload a previously downloaded device or device group to the portal:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), 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/admin/settings/device`

   > **See Also:**
   >
   > WebCenter Portal Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. On the **Device Groups** or **Devices** tab, click **Upload**.
3. Use the Upload Devices/Device Groups dialog to locate the `.aar` file on your system. Select:
   - **Look on My Computer** to upload an archive file from your local file system. Click **Browse** to locate the file.
   - **Look on WebCenter Portal Server** to upload an archive file from a remote server file system. In the field, enter the location on the server where the file is located.

     When you download an archive file, it is saved to the default path *`DOMAIN_HOME`*`/WC_Archives`, where *`DOMAIN_HOME`* refers to the domain location where WebCenter Portal is installed.

     See also Downloading an Asset in *Building Portals with Oracle WebCenter Portal*.
4. Click **Upload**.
5. If the archive already exists in WebCenter Portal, click **Yes** to confirm that you want to replace the device group or device with the contents of the archive file.
6. Click **OK** in the resulting success dialog.

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 Previewing a Mobile Device Variant of a Page in *Building Portals with Oracle WebCenter Portal*.

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 Porta

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.

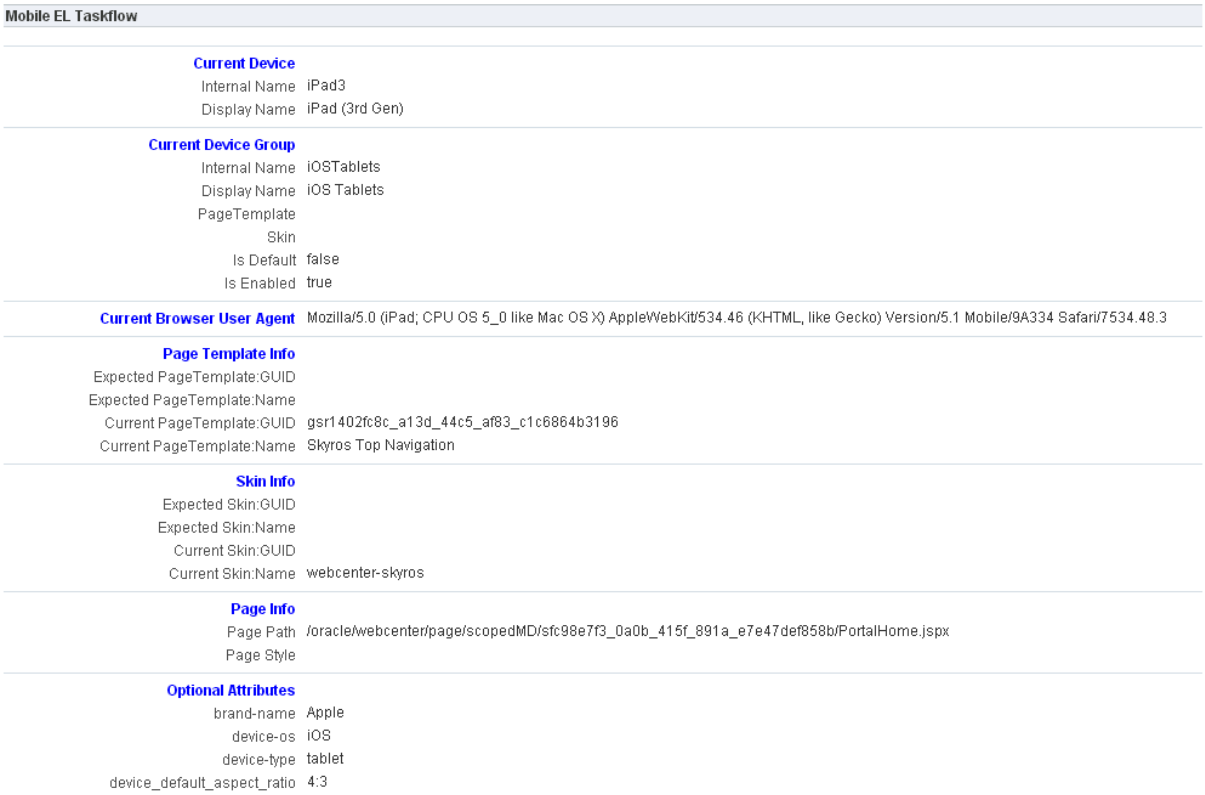
If You Need Information About the Requesting Device

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, 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.

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 EL Expressions Related to Device Settings in *Developing for Oracle WebCenter Portal*. 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 43-10 shows the output from a task flow created with this sample code.

Figure 43-10 Output from Sample Task Flow



Mobile EL Taskflow

Current Device
Internal Name: iPad3
Display Name: iPad (3rd Gen)

Current Device Group
Internal Name: iOSTablets
Display Name: iOS Tablets
PageTemplate:
Skin:
Is Default: false
Is Enabled: true

Current Browser User Agent: Mozilla/5.0 (iPad; CPU OS 5_0 like Mac OS X) AppleWebKit/534.46 (KHTML, like Gecko) Version/5.1 Mobile/9A334 Safari/7534.48.3

Page Template Info
Expected PageTemplate:GUID:
Expected PageTemplate:Name:
Current PageTemplate:GUID: gsr1402fc8c_a13d_44c5_af83_c1c6864b3196
Current PageTemplate:Name: Skyros Top Navigation

Skin Info
Expected Skin:GUID:
Expected Skin:Name:
Current Skin:GUID:
Current Skin:Name: webcenter-skyros

Page Info
Page Path: /oracle/webcenter/page/scopedMD/sfc98e7f3_0a0b_415f_891a_e7e47def858b/PortalHome.jspx
Page Style:

Optional Attributes
brand-name: Apple
device-os: iOS
device-type: tablet
device_default_aspect_ratio: 4:3

44

Customizing Task Flows

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 Adding Custom Actions to a Task Flow in *Developing for Oracle WebCenter Portal*.

> **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 All` or `Portal Server-Manage Configuration`

Topics:

- About Task Flow Customization at the Application Level
- Customizing Task Flows at the Application Level
- Removing Task Flow Customizations

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 manager can customize the task flow on the portal-level **Task Flow Editor** system page. See Working with Task Flows in *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 Removing All Page Customizations from a System Page and Removing Task Flow Customizations, respectively.

Customizing Task Flows at the Application Level

This section describes how to perform task flow customizations for WebCenter Portal, at the application level.

To perform application-wide task flow customizations through the **Task Flow Editor** system page:

1. On the **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **System Pages**.

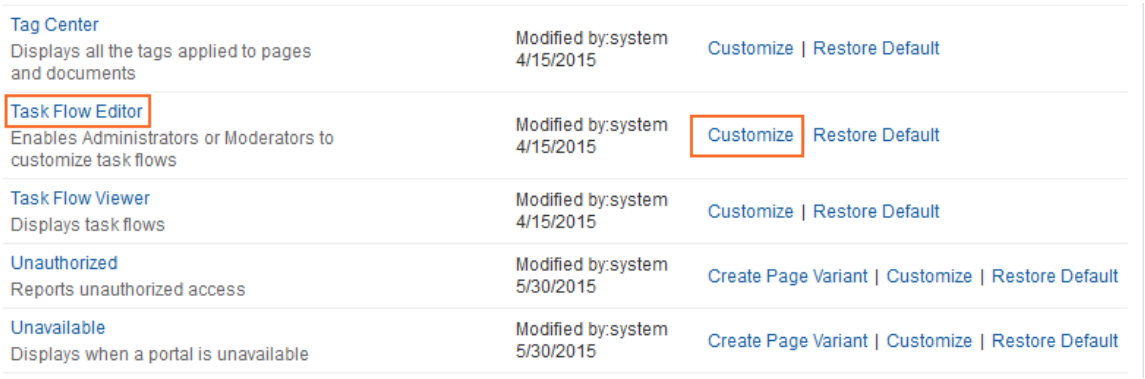
   You can also enter the following URL in your browser to navigate directly to the **System Pages** page:

   `http://host:port/webcenter/portal/admin/settings/systempages`

   > **See Also:**
   >
   > Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

2. Click the **Customize** link next to the **Task Flow Editor** system page (Figure 44-1) to open it in Structure view in the page editor.

Figure 44-1 Customize Link Next to the Task Flow Editor System Page

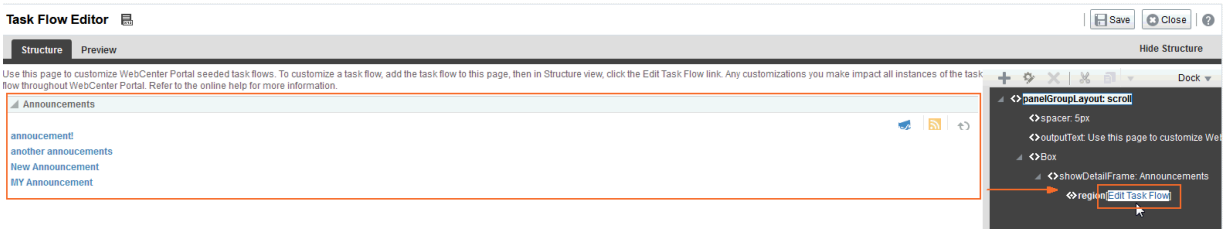


3. Add a task flow that you want to edit to the **Task Flow Editor** page (Figure 44-2).
   a. Click on the page to activate it.
   b. Click the **Add content into the selected component** icon (the Add icon on the right toolbar).

      The Add Content dialog opens.

   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 Adding a Component to a Page in *Building Portals with Oracle WebCenter Portal*.

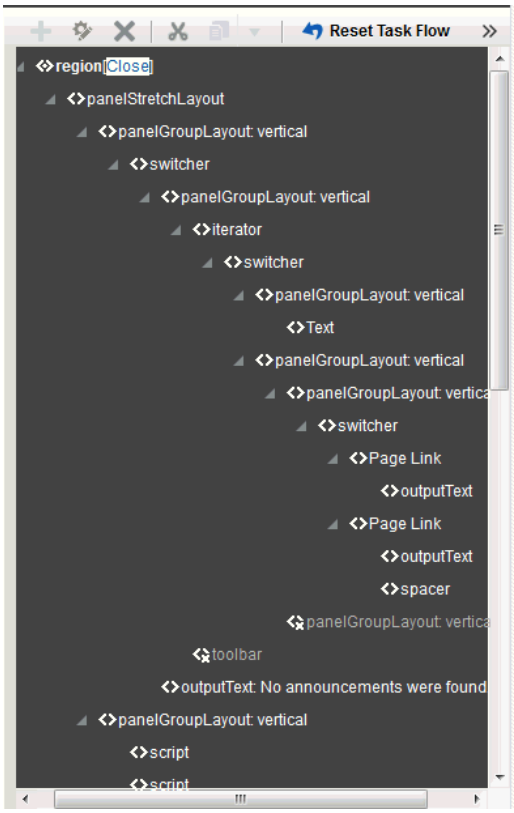
Figure 44-2 Edit Task Flow Link Next to a Region Tag



4. Click the **Edit Task Flow** link next to the task flow you want to customize (Figure 44-2).
5. In the Confirm Task Flow Edit dialog, click **Edit**.

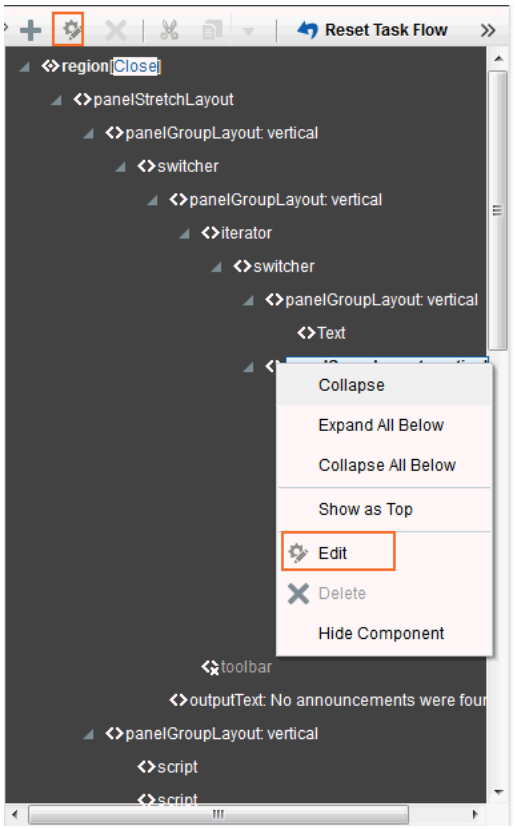
   Structure view zooms into the source code hierarchy of the task flow being edited (Figure 44-3)

Figure 44-3 Zoomed-In View of Task Flow



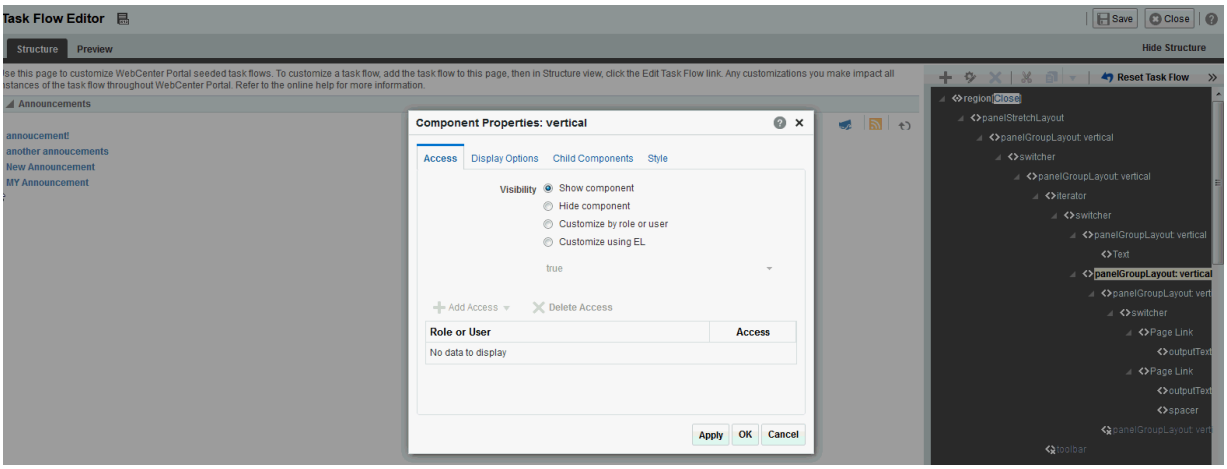
6. Set the properties of a task flow element by clicking it in the Task Flow Editor, then click the **Show the properties of *region*** icon. Alternatively, right-click the `region` and select **Edit** (Figure 44-4).

Figure 44-4 Selected Task Flow Element on a Page in Structure View



The Component Properties dialog opens (Figure 44-5).

Figure 44-5 Component Properties Dialog



7. Make your changes to the element's properties. Click the **Help** icon for descriptions of the parameters for the component you are editing

> **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.

8. 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.

9. Click **Save** then **Close** to exit Composer.

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 **Settings** page (see Accessing the Settings Pages in WebCenter Portal Administration), click **System Pages**.

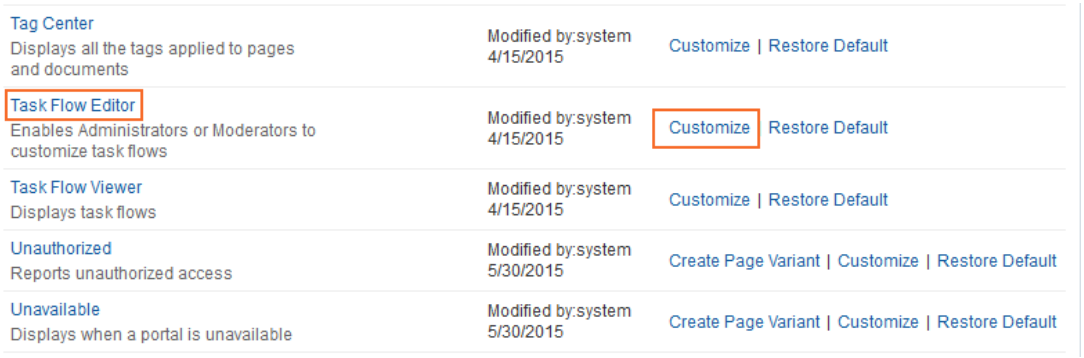
   You can also enter the following URL in your browser to navigate directly to the **System Pages** page:

   `http://host:port/webcenter/portal/admin/settings/systempages`

> **See Also:**
>
> Pretty URLs in *Building Portals with Oracle WebCenter Portal*.

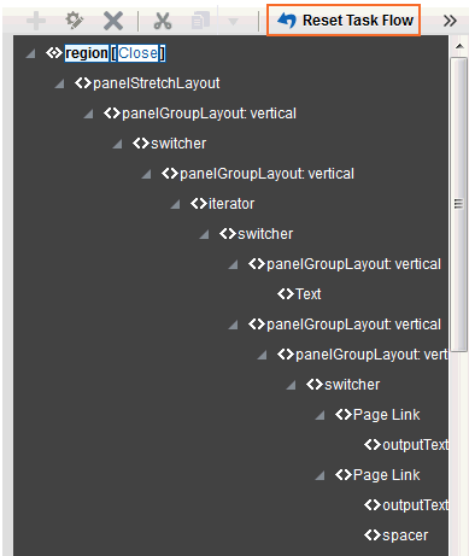
2. Click the **Customize** link next to the **Task Flow Editor** system page (Figure 44-6) to open it in page edit mode.

Figure 44-6 Customize Link Next to a System Page



3. In the Structure view toolbar, click **Reset Task Flow**.

Figure 44-7 Reset Task Flow Option for a Selected Element



4. In the Reset Task Flow dialog, click **Reset Task Flow** to confirm the action.

45
Analyzing Portal Usage

Use the Analytics service to monitor real-time usage and activity reporting for your portal.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

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. For more information about the Analytics schema and how to manage the Analytics service backend, see Managing Analytics.

This chapter describes how to add Analytics task flows to portal pages at runtime. To learn how to add Analytics task flows at design time using Oracle JDeveloper, see *Developing for Oracle WebCenter Portal*.

> **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 About Application Roles and Permissions.

Topics:

- About the Analytics Task Flows and Service
- About the Analytics Administration Page
- Working with Analytics Task Flows

About the Analytics Task Flows and Service

The Analytics service allows WebCenter Portal administrators and portal managers 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.

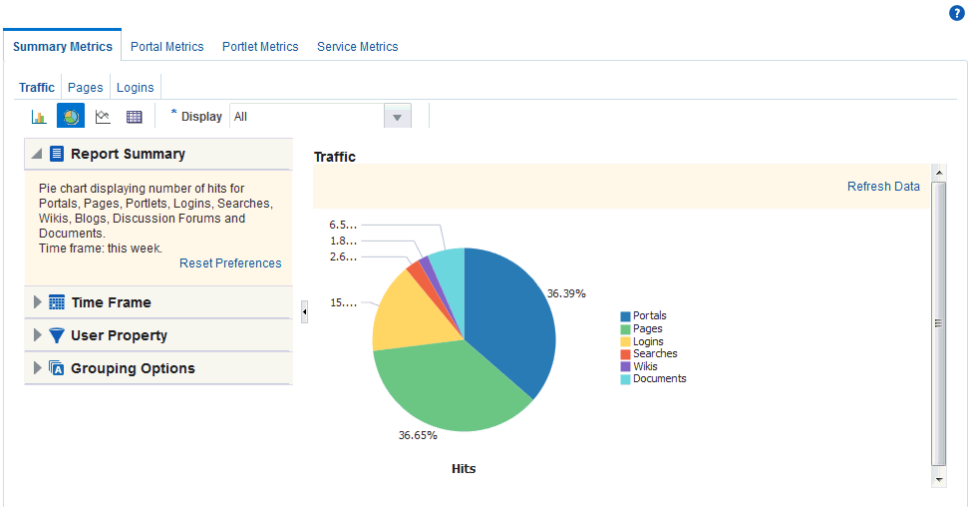
> **Note:**
>
> Profile information is cached meaning that changes to a user profile are not visible in reports until the cache is updated. The default cache time is 60 minutes, but this value can be changed by your administrator.

About the Analytics Administration Page

An *analytics console* that displays metrics for the entire Oracle 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 45-1 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.

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:

- Understanding Analytics Task Flows
- Adding Analytics Task Flows to a Page
- Customizing Analytics Reports
- Personalizing Your Analytics Report
- Setting Analytics Task Flow Properties

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:

- WebCenter Traffic
- Page Traffic (Administrator)
- Login Metrics (System Administrator)

Portal Analytics:

- Portal Traffic (System Administrator)
- Portal Response Time (System Administrator)

Portlet Analytics:

- Portlet Traffic (Administrator)
- Portlet Instance Traffic (Administrator)
- Portlet Response Time (Administrator)
- Portlet Instances Response Time (Administrator)

Service Analytics:

- Search Metrics
- Document Metrics (System Administrator)
- Wiki Metrics (System Administrator)

- Blog Metrics (System Administrator)
- Discussion Forum Metrics (System Administrator)

> **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 Customizing Analytics Reports and Personalizing Your Analytics Report .

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 WebCenter Traffic in *Building Portals with Oracle WebCenter Portal*.

Page Traffic (Administrator)

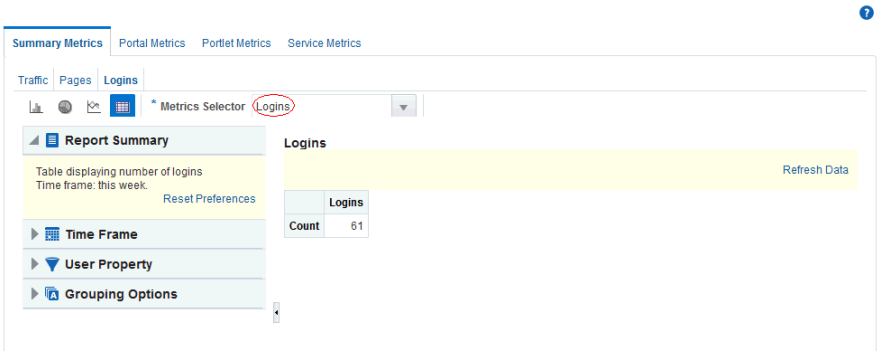
The Page Traffic task flow displays the number of page hits and the number of unique users that have visited any portal page. Use this task flow to quickly see the most visited pages (top pages) and/or the least visited pages (bottom pages). For more information, see Page Traffic (Administrator) in *Building Portals with Oracle WebCenter Portal*.

Login Metrics (System Administrator)

The Login task flow (Figure 45-2) 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 45-2 Analytics Task Flow - Login Metrics

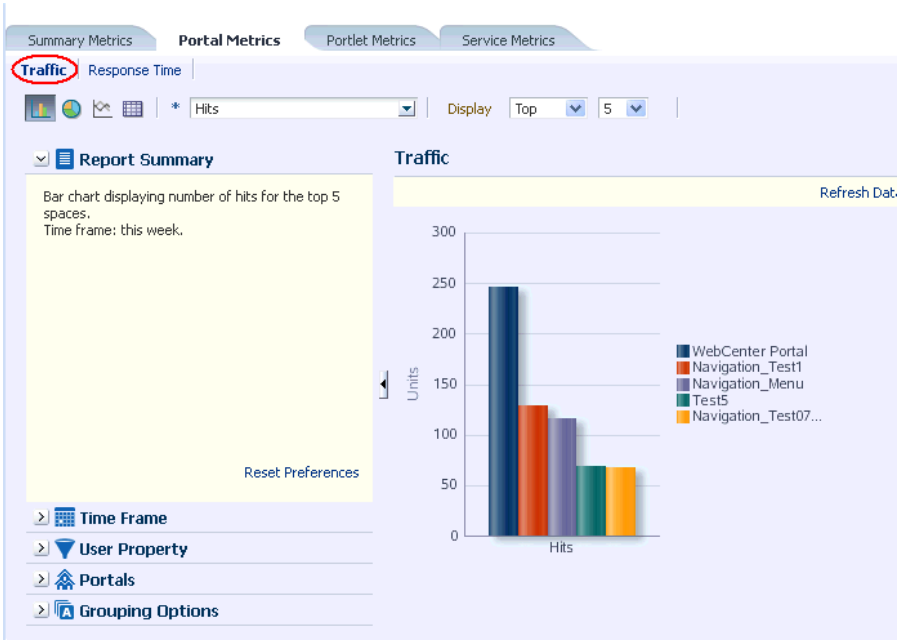


Portal Traffic (System Administrator)

The Portal Traffic task flow (Figure 45-3) 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 45-3 Analytics Task Flow - Portal Traffic

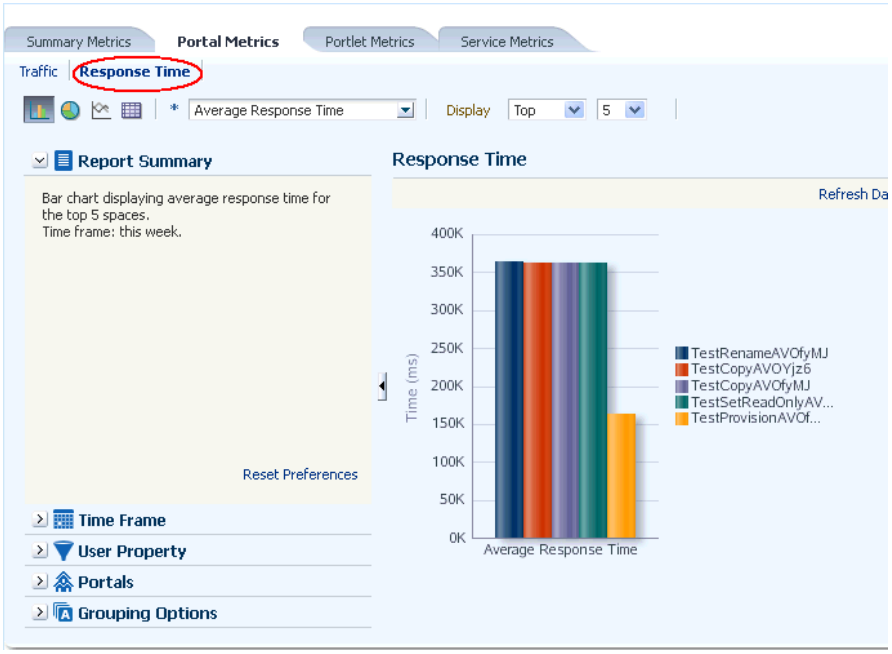


> **Note:**
>
> The Home portal is not included in the data.

Portal Response Time (System Administrator)

The Portal Response Time task flow (Figure 45-4) 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 45-4 Analytics Task Flow - Portal Response Time



> **Note:**
>
> The Home Portal is not included in the data.

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 Portlet Traffic (Administrator) in *Building Portals with Oracle WebCenter Portal*.

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 Portlet Instance Traffic (Administrator) in *Building Portals with Oracle WebCenter Portal*.

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 Portlet Response Time (Administrator) in *Building Portals with Oracle WebCenter Portal*.

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 Portlet Instances Response Time (Administrator) in *Building Portals with Oracle WebCenter Portal*.

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 Search Metrics in *Building Portals with Oracle WebCenter Portal*.

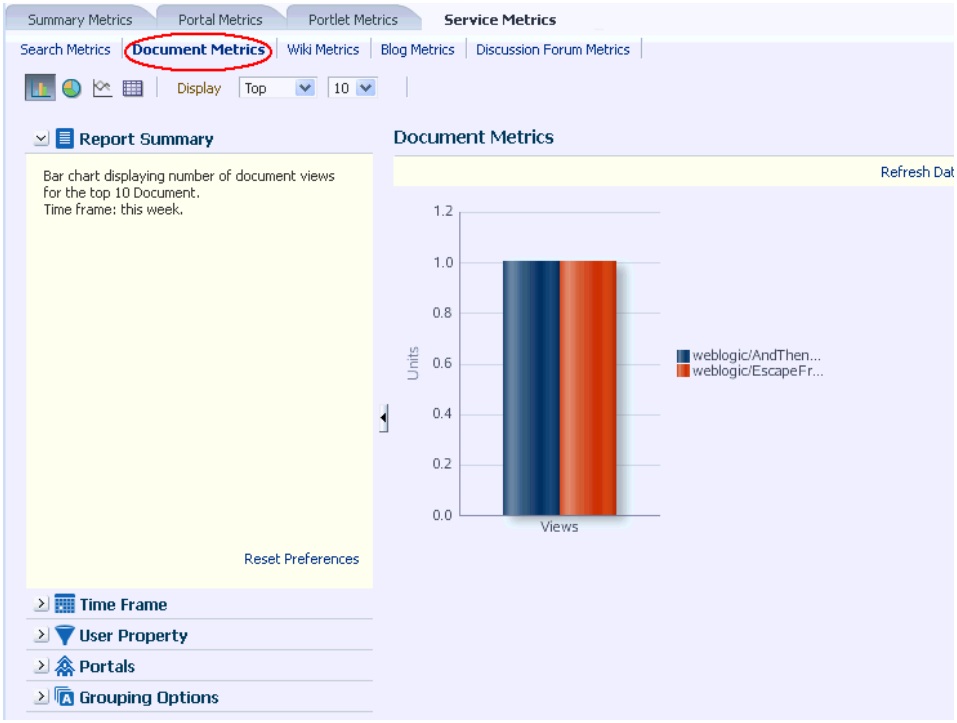
Document Metrics (System Administrator)

The Document Metrics task flow (Figure 45-5) 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.

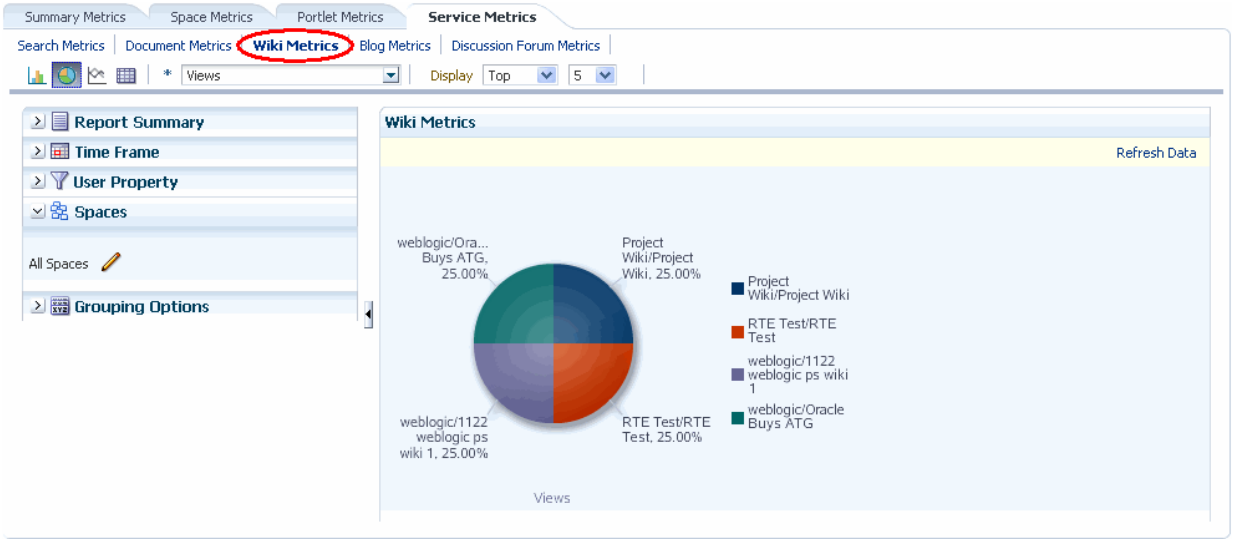
Figure 45-5 Analytics Task Flow - Document Metrics



Wiki Metrics (System Administrator)

The Wiki Metrics task flow (Figure 45-6) 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.

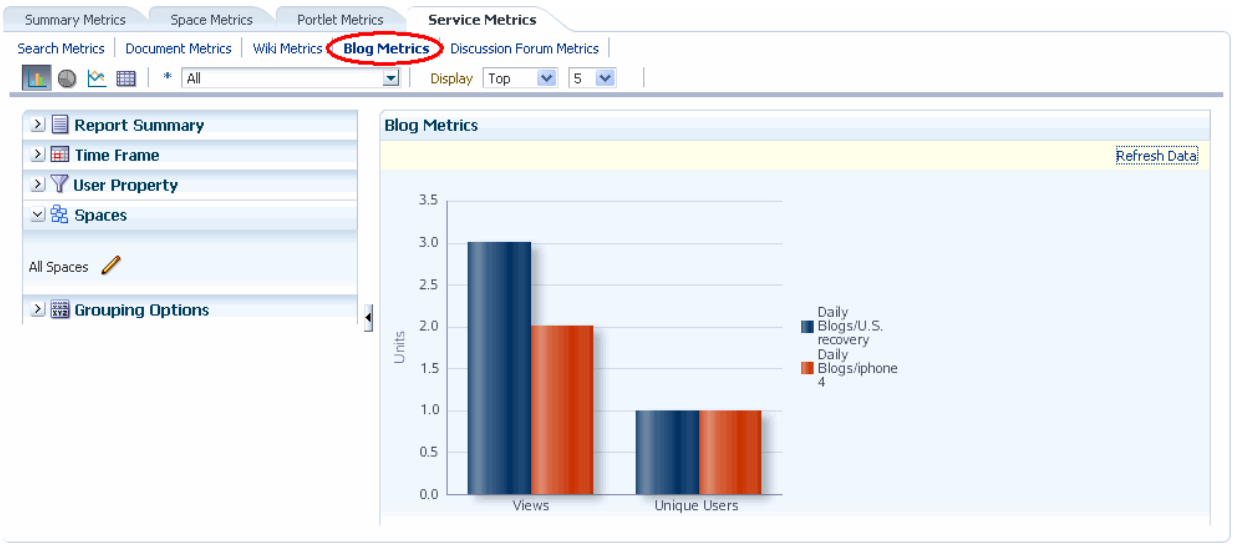
Figure 45-6 Analytics Task Flow - Wiki Metrics



Blog Metrics (System Administrator)

The Blog Metrics task flow (Figure 45-7) 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.

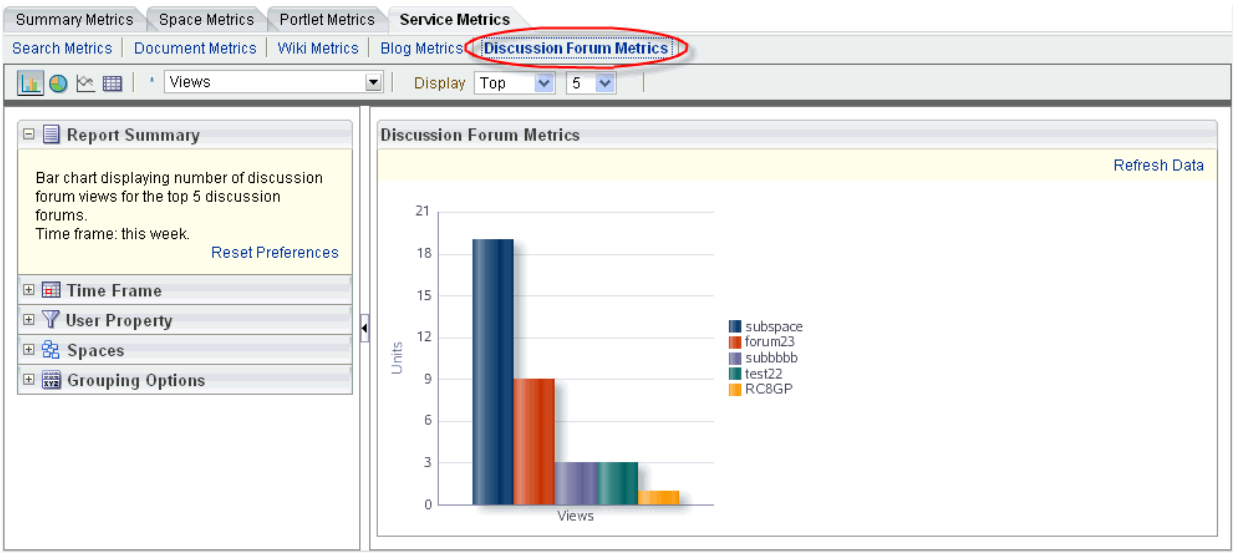
Figure 45-7 Analytics Task Flow - Blog Metrics



Discussion Forum Metrics (System Administrator)

The Discussion Forum Metrics task flow (Figure 45-8) 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.

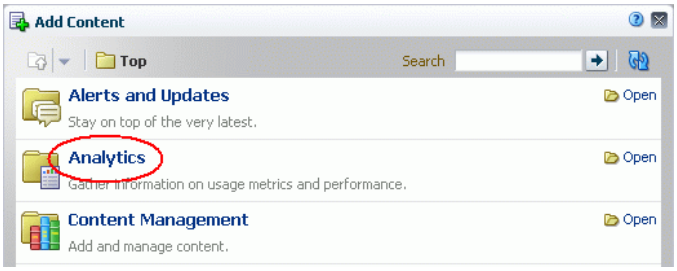
Figure 45-8 Analytics Task Flow - Discussion Metrics



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 Adding Analytics to a Portalin *Building Portals with Oracle WebCenter Portal*). The process varies only in where you find these task flows in the resource catalog. All the Analytics task flows are under the **Analytics** folder.

Figure 45-9 Analytics Folder in Resource Catalog



> **Note:**
>
> When you add an Analytics task flow to a page in a portal, it displays information only for that portal, not for all portals.

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 Customizing Analytics Reports in *Building Portals with Oracle WebCenter Portal*.

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:

- Report Display Options
- Query Options

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 45-1 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 45-1 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

Table 45-1 (Cont.) Display Options for the Analytics Task Flows

Selected Metrics	Group By Option	Bar	Pie	Line	Table
Multiple metrics Login Traffic task flow	Both*	N	N	N	Y

* The grouping option **Both** is available only for the Login Traffic task flow.

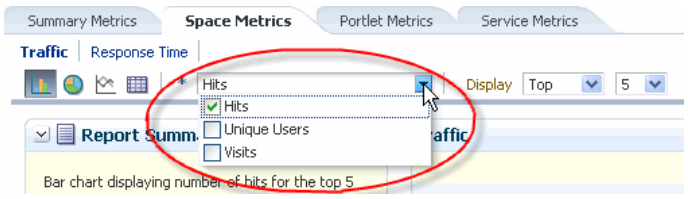
Metrics

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: Portals, 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 45-10 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.

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).

- **Portals**

When Analytics task flows display in the Home portal or on a business role page, you can choose which portals to include in your report. When Analytics task flows are used within a particular portal, only metrics only for that portal display; the Portals option is unavailable (grayed out).

To specify the portals to include in your report, click the **Portal Filter** icon to display the Specify Portals popup. Select the portals you want to include in your report, using Ctrl+click and Shift+click to select multiple portals.

This option is not available with the Traffic, Logins, or Search Metrics task flows.

- **Grouping Options**

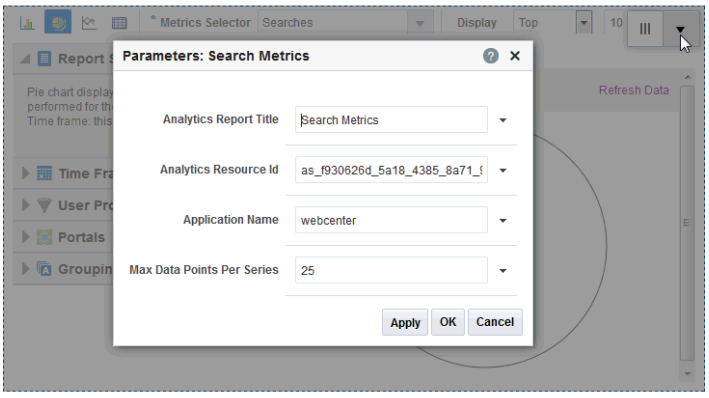
  Enables you to select an option by which to group the metrics in your report. You can group by a time interval (Hour, Day, Week, Month, or Year), a user property, or, with the Logins task flow, both.

> **Note:**
>
> This setting affects the available display options for the report (see Table 45-1).

Setting Analytics Task Flow Properties

The Analytics service task flows have associated properties, which users with sufficient privileges can access through the task flows' **View Actions** menu. For example, select **Parameters** to show the Parameters dialog (Figure 45-11).

Figure 45-11 Analytics Task Flow: Parameters Dialog



The following sections provide information about properties of the Analytics service task flows and describe the task flow parameters:

- About the Analytics Service Task Flow Properties
- Analytics Service Task Flow Parameters

About the Analytics Service Task Flow Properties

When you edit a page, the **View Actions** menu appears in the toolbar of the Analytics task flows when you click the task flow. The **View Actions** menu provides access to the properties dialogs: Parameters, Access, Display Options, Style, and Content Style.

- Parameters control the default task flow content. For descriptions of each parameter, see Analytics Service Task Flow Parameters. Parameters can be wired to events, and can be used facilitate the wiring of the task flow to page parameters and page definition variables. For more information, see Wiring Pages, Task Flows, Portlets, and UI Components in *Building Portals with Oracle WebCenter Portal*.
- Access settings show or hide the component to specific roles, users, or groups. For more information, see Setting Component Access in *Building Portals with Oracle WebCenter Portal*.
- The Display Options, Style, and Content Style properties affect the appearance and behavior of the task flow for all users. These properties are common to all task flows. For more information, see Modifying Components in *Building Portals with Oracle WebCenter Portal*.

The Parameters and Display Options dialogs 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 ▾ icon next to a property, then select **Expression Builder** to open the editor.

> **Note:**
>
> When you enter EL in the Display Options dialog, the parser reports an error only if it detects invalid syntax, such as a missing closing bracket. Validation is performed only on syntax, not on the expression value. Generic Display Options are those cataloged in Setting Component Display Options Properties in *Building Portals with Oracle WebCenter Portal*.
>
> EL validation is not performed on non-generic display options.

If you need EL assistance, an application developer can provide an EL expression; see Expression Language Expressions in *Developing for Oracle WebCenter Portal*.

Analytics Service Task Flow Parameters

Table 45-2 describes the parameters that are unique to the Analytics service task flows.

Table 45-2 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 in the Display Options dialog. Changing the Text value has no effect on Analytics task flows. • You cannot change the report titles in the Analytics console.

Table 45-2 (Cont.) Analytics Task Flow Parameters

Parameter	Description
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 IX
Appendixes

This part of *Administering Oracle WebCenter Portal* provides appendixes with supporting information for the chapters in this guide.

- Managing Announcements and Discussions
- Oracle WebCenter Portal Configuration
- Third-Party Product Support
- Migrating Wiki Content to WebCenter Portal
- Migrating Folders_g to FrameworkFolders
- Troubleshooting WebCenter Portal

A

Managing Announcements and Discussions

Configure and manage announcements and discussions for WebCenter Portal.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

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.

Any configuration changes that you make postdeployment are stored in the MDS metatdata store as customizations.

> **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.
>
> For troubleshooting issues, see 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 granted through WebCenter Portal Administration.
>
> For more information about roles and permissions, see Understanding Administrative Operations, Roles, and Tools.

> **Important:**
>
> Oracle supports the embedded discussions server from Jive Software. Oracle supports the features in Jive that are available through the WebCenter Portal task flows. Any custom development using APIs in the Jive WebService layer are subject to review by Oracle and cannot be supported. There are other features that Jive Software delivers as part of the discussions server that Oracle does not recommend and cannot support. Documentation for Jive Forums is included for reference only. Jive software installations and upgrades outside of the WebCenter Portal product installation are not supported.

Topics:

- About Discussions Server Connections
- Discussions Server Prerequisites
- Registering Discussions Servers
- Choosing the Active Connection for Discussions and Announcements
- Modifying Discussions Server Connection Details
- Deleting Discussions Server Connections
- Setting Up Discussions Defaults
- Setting Up Announcements Defaults
- Testing Discussions Server Connections
- Granting Administrator Permissions on the Discussions Server
- Granting Administrator Role on the Discussions Server
- Configuring Discussion Forum Options for WebCenter Portal

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. 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.

Discussions Server Prerequisites

This section includes the following subsections:

- Discussions Server - Installation
- Discussions Server - Configuration
- Discussions Server - Security Considerations
- Discussions Server - Limitations

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.

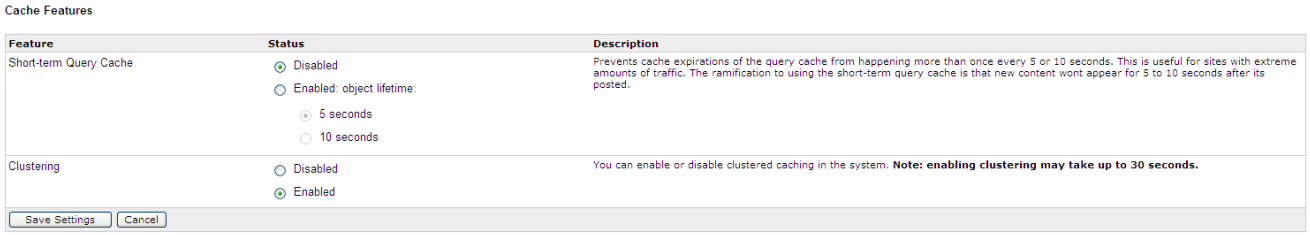
The Oracle Fusion Middleware Configuration Wizard automatically creates managed servers in the domain to host the selected WebCenter Portal components. For information, see Selecting the Configuration Templates for Oracle WebCenter Portal in *Installing and Configuring Oracle WebCenter Portal*.

You can set up WebCenter Portal's Discussion Server for high availability if you have installed `WC_Collaboration` domain in a clustered environment.

To set up WebCenter Portal's Discussion Server for high availability:

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. Go to the Cache Settings page (click the **System** link at the top of the page and select **Cache Settings**), then scroll down to the Cache Features section, and select **Enabled** to enable clustering.

Figure A-1 Cache Features - Clustering



> **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.

Discussions Server - Configuration

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 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 WLST commands described in the table.

Table A-1 Discussions Server WLST Commands

WLST Command	Purpose	Link in *WebCenter WLST Command Reference Reference*
`getDiscussionsServerProperty`	Return discussion server property values	See getDiscussionsServerProperty
`setDiscussionsServerProperty`	Set discussion server properties	See setDiscussionsServerProperty
`removeDiscussionsServerProperty`	Remove currently set discussion server property values	See removeDiscussionsServerProperty
`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 addDiscussionsServerAdmin

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 Discussions and Announcements in *WebCenter WLST Command Reference Reference*.

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 by following the steps for the required topology in Configuring Web Services Security.

- 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 Configuring SSL and Configuring Web Services 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 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 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 Migrating the Discussions Server to Use an External LDAP.

- Oracle recommends that you install and configure a single sign-on solution to avoid users having to log in twice when accessing WebCenter Portal's Discussion Server and other WebCenter Portal components. You can configure the discussions 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 more information about single sign-on solutions, see Configuring Single Sign-On. For additional discussions-specific configuration instructions for Oracle Access Manager (OAM), see 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 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.
- Admin access to WebCenter Portal's Discussion Server Admin Console (`/owc_discussions/admin`) is disabled by default. You can enable the access by setting the system property `jiveAllowUnsecureAdminAccess=true`.

> **Note:**
>
> To set the system property `jiveAllowUnsecureAdminAccess`, follow the steps:
>
> 1. Create the `setUserOverrides.sh` file in the `DOMAIN_HOME/bin`.
> 2. Add the following to `EXTRA_JAVA_PROPERTIES` entry, if server name is `WC_Collaboration`.
>
>    ```
>    jiveAllowUnsecureAdminAccess=true
>    ```
>
>   For example,
>
>    ```
>    if [ "${SERVER_NAME}" = "WC_Collaboration" ] ; then
>      echo "Customizing EXTRA_JAVA_PROPERTIES for
>    SERVER_NAME ${SERVER_NAME}"
>       EXTRA_JAVA_PROPERTIES="-
>    DjiveAllowUnsecureAdminAccess=true
>      ${EXTRA_JAVA_PROPERTIES}"
>      export EXTRA_JAVA_PROPERTIES
>    fi
>    ```

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`.

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 topics:

- Registering Discussions Servers Using Fusion Middleware Control
- Registering Discussions Servers Using WLST

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.
2. From the **WebCenter Portal** menu, select **Settings** > **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 A-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 A-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.
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.

Table A-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.
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.

Table A-3 (Cont.) Discussion and Announcement Connection - Connection Details

Field	Description
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.

7. Configure advanced options for the discussion and announcement connection.

Table A-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 the table.

Table A-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.

Table A-5 (Cont.) Additional Discussion Connection Properties

Additional Connection Property	Description
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 A-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 is deployed.

Some additional configuration is required to use discussions and announcements in WebCenter Portal. For details, see Configuring Discussion Forum Options for WebCenter Portal.

Registering Discussions Servers Using WLST

Use the WLST command `createDiscussionForumConnection` to create a discussions server connection. For command syntax and examples, see createDiscussionForumConnection in *WebCenter WLST Command Reference 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 Modifying Discussions Server Connection Details Using WLST.

> **Note:**
>
> To start using the new (active) connection, you must restart the managed server on which WebCenter Portal is deployed.

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:

- Choosing the Active Connection for Discussions and Announcements Using Fusion Middleware Control
- Choosing the Active Discussion for Discussions and Announcements Using WLST

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 WebCenter Portal.
2. From the **WebCenter Portal** menu, select **Settings** > **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 is deployed.

Choosing the Active Discussion for Discussions and Announcements Using WLST

Use the WLST command `setDiscussionForumConnection` with `default=true` to activate an existing connection. For command syntax and examples, see setDiscussionForumConnection in *WebCenter WLST Command Reference 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 removeDiscussionForumServiceProperty in *WebCenter WLST Command Reference Reference*.

> **Note:**
>
> To start using the new (active) connection you must restart the managed server on which WebCenter Portal is deployed.

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:

- Modifying Discussions Server Connection Details Using Fusion Middleware Control
- Modifying Discussions Server Connection Details Using WLST

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.
2. From the **WebCenter Portal** menu, select **Settings** > **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 A-3 and Table A-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 is deployed.

Modifying Discussions Server Connection Details Using WLST

Use the WLST command `setDiscussionForumConnection` to edit connection details. For command syntax and examples, see setDiscussionForumConnection in *WebCenter WLST Command Reference Reference*.

To set additional parameters, use the `setDiscussionForumConnectionProperty` command. For more information, see setDiscussionForumConnectionProperty in *WebCenter WLST Command Reference Reference*.

> **Note:**
>
> To start using the updated (active) connection you must restart the managed server on which WebCenter Portal is deployed.

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:

- Deleting a Discussions Server Connection Using Fusion Middleware Control
- Deleting a Discussions Server Connection Using WLST

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.
2. From the **WebCenter Portal** menu, select **Settings** > **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 is deployed.

Deleting a Discussions Server Connection Using WLST

Use the WLST command `deleteConnection` to remove a connection. For command syntax and examples, see deleteConnection in *WebCenter WLST Command Reference Reference*.

Ensure that another connection is marked active; otherwise, the tool is disabled. For the changes to take effect, you must restart the managed server on which WebCenter Portal is deployed.

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 Publishing Portal Mail in a Discussion Forum in *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 command syntax and examples, see setDiscussionForumServiceProperty in *WebCenter WLST Command Reference Reference*.

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 Discussions and Announcements in *WebCenter WLST Command Reference Reference*.

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.

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 addDiscussionsServerAdmin and addDiscussionsCategoryAdmin in *WebCenter WLST Command Reference Reference*.

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 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.

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 addDiscussionsServerAdmin in *WebCenter WLST Command Reference Reference*.

Granting the Discussions Server Administrator Role Using the Admin Console

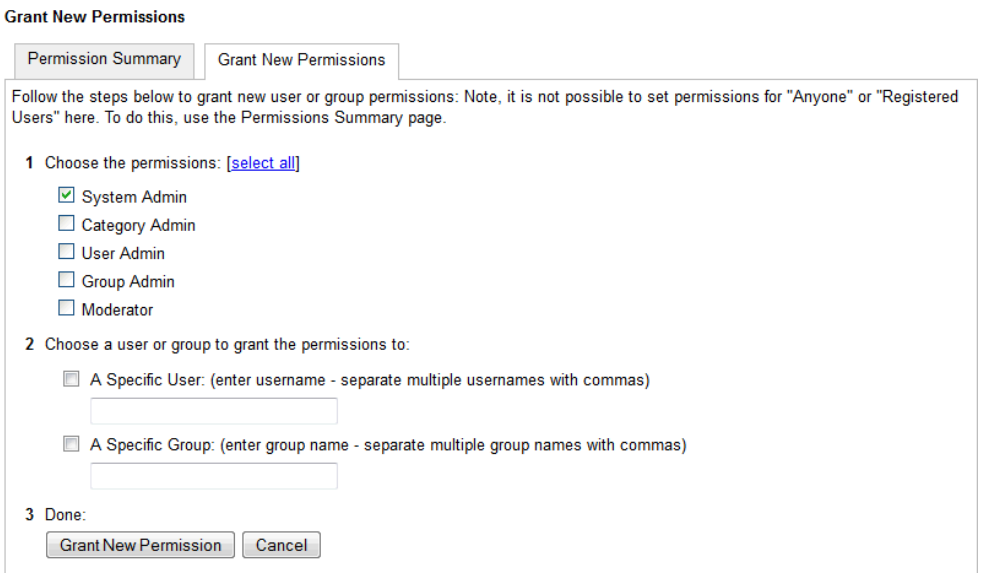
To grant the administrator role for WebCenter Portal's Discussion Server to a nondefault user:

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 A-3 Granting the Administrator Role on WebCenter Portal's Discussion Server



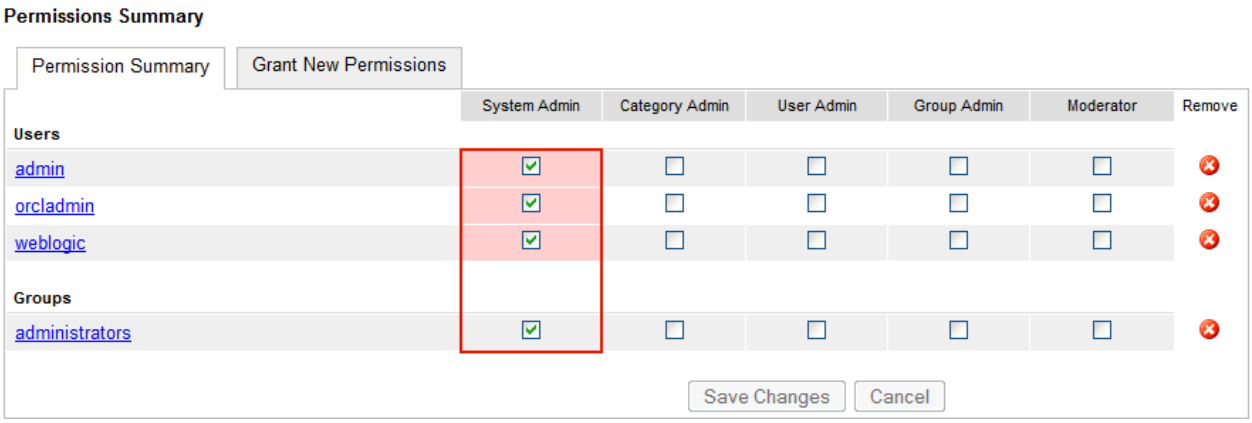
Revoking the Discussions Server Administrator Role

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.
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 A-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.

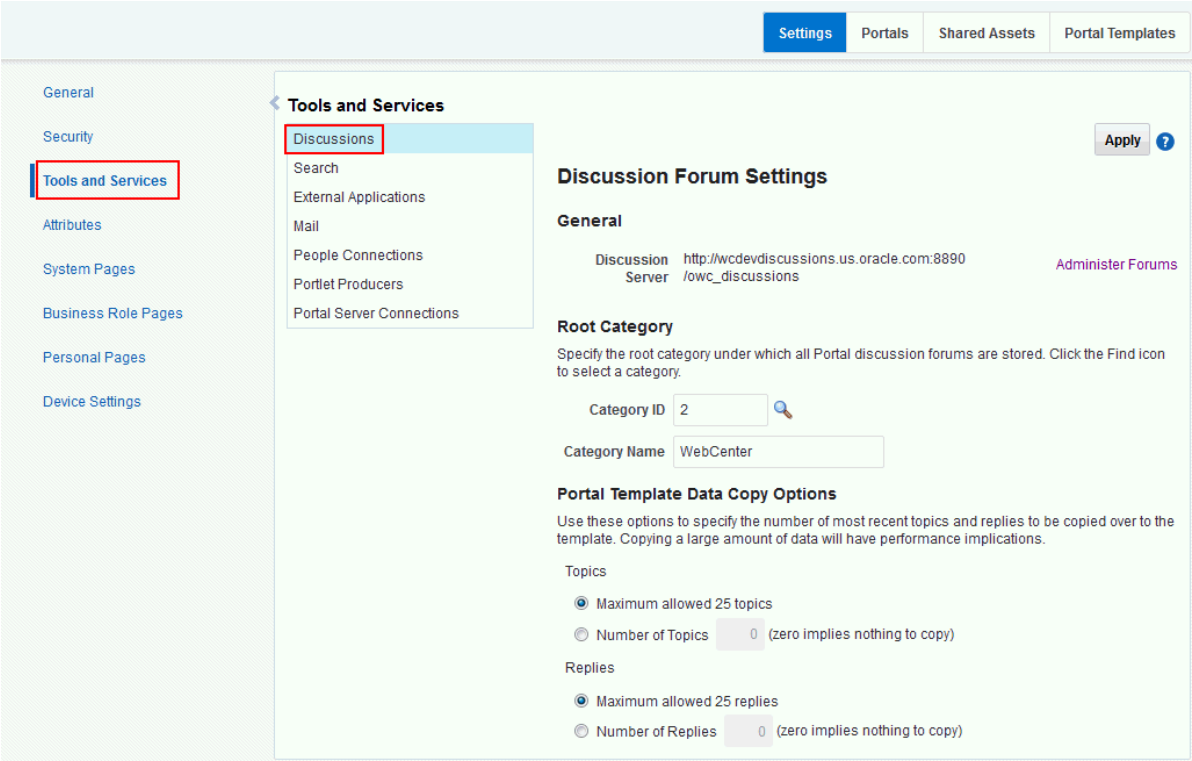
Configuring Discussion Forum Options for WebCenter Portal

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` (`Portal Server – Manage All`) permissions.

As an administrator, you are responsible for setting discussion forum options for the entire application through the Discussion Forum Settings page in WebCenter Portal Administration.

Figure A-5 Setting Discussion Forum Options



From the Discussion Forum Settings page you configure discussions-related setting, as well as access the discussions server administration pages:

- Accessing the Discussions Server Admin Console
- Specifying Where Discussions and Announcements are Stored on the Discussions Server
- 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 Registering Discussions Servers.

Accessing the Discussions Server Admin Console

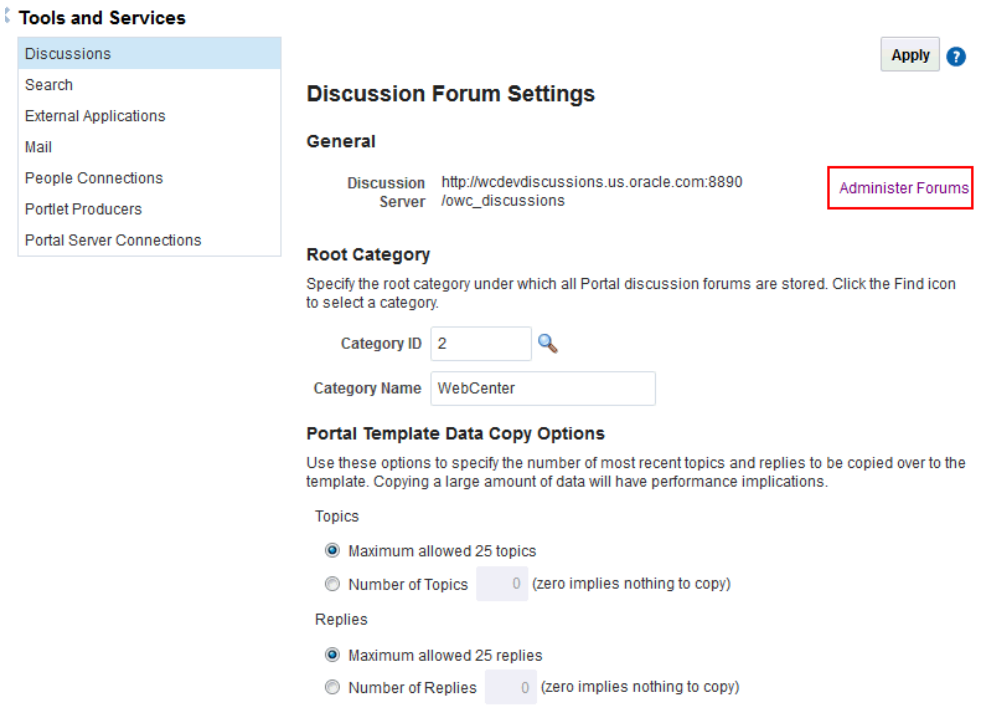
For convenience, you can access the discussions server's Admin Console, a web-based tool for configuring and managing discussion forums, from WebCenter Portal's Administration pages. In the discussions server's Admin Console, you can navigate all categories and forums and edit their properties, create new categories and forums, as well as set cache, security, and various other properties for the discussions server.

Admin access to WebCenter Portal's Discussion Server Admin Console (`/owc_discussions/admin`) is disabled by default. You can enable the access by setting the system property `jiveAllowUnsecureAdminAccess=true`

> **Note:**
>
> To set the system property `jiveAllowUnsecureAdminAccess`, follow the steps:
>
> 1. Create the `setUserOverrides.sh` file in the `DOMAIN_HOME/bin`.
> 2. Add the following to `EXTRA_JAVA_PROPERTIES` entry, if server name is `WC_Collaboration`.
>
>    ```
>    jiveAllowUnsecureAdminAccess=true
>    ```
>
>   For example,
>
>    ```
>    if [ "${SERVER_NAME}" = "WC_Collaboration" ] ; then
>      echo "Customizing EXTRA_JAVA_PROPERTIES for SERVER_NAME $
>    {SERVER_NAME}"
>       EXTRA_JAVA_PROPERTIES="-DjiveAllowUnsecureAdminAccess=true
>      ${EXTRA_JAVA_PROPERTIES}"
>      export EXTRA_JAVA_PROPERTIES
>    fi
>    ```

1. Log on to WebCenter Portal, and access WebCenter Portal Administration by selecting **Administration** from the **Portals** menu, then clicking **Settings**.
2. Click **Tools and Services**, and then select **Discussions**.
3. Click **Administer Forums**.

Figure A-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 **Administer Forums** 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).

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 recommends the following:

- Choose a category that is dedicated to WebCenter Portal. There may be conflicts when multiple Oracle 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.

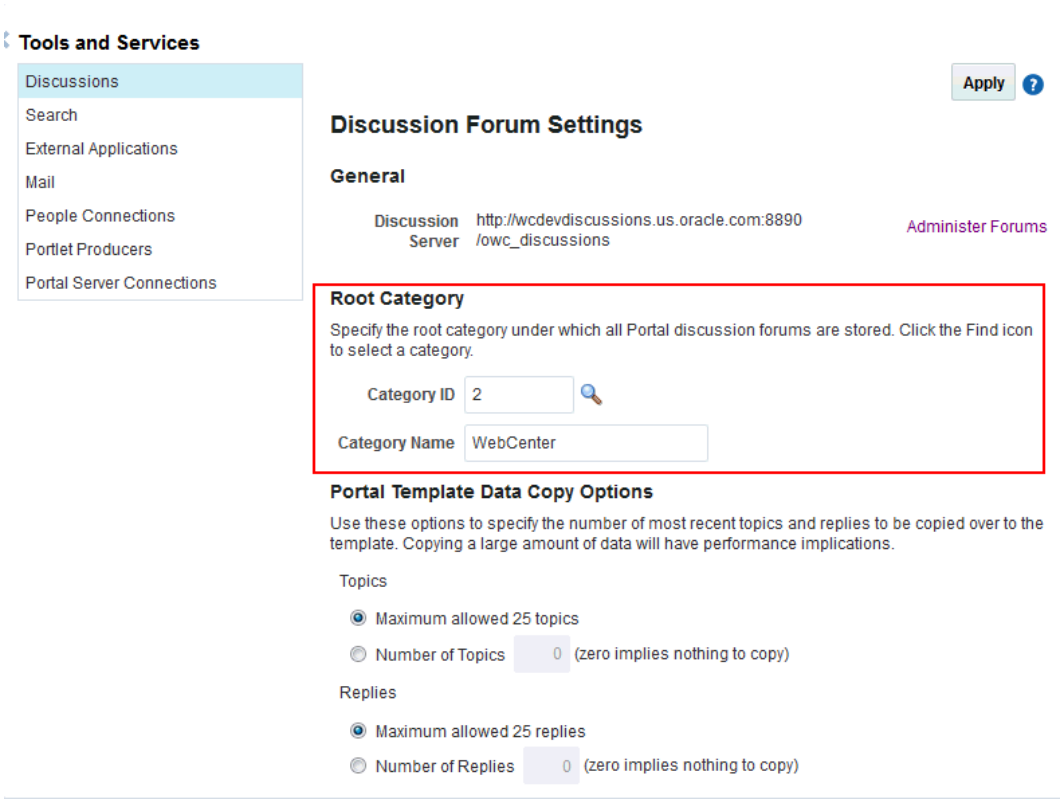
You can retain existing discussions in a portal template saved with the data copy option. For example, in the WebCenter Portal Administration **Tools and Services - Discussions** page, enter the number (between 1 and 25) of most recent topics and replies to be copied over to the template.

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.
2. Click **Tools and Services**, and then select **Discussions**.

Figure A-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.

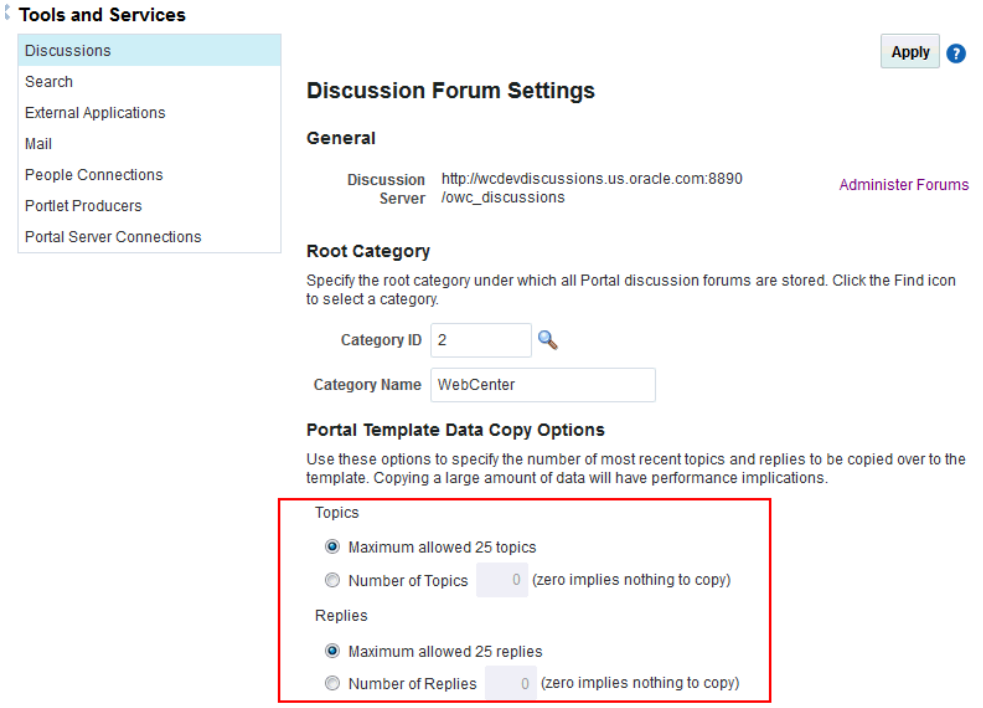
   To create a new category, click **Create Category**. You must have system administrator permissions on the discussions server to create new categories.
4. Click **Apply** to save the settings.

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.
2. Click **Tools and Services**, and then select **Discussions**.

Figure A-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.

B

Oracle WebCenter Portal Configuration

Learn about the two main configuration files for WebCenter Portal, `adf-config.xml` and `connections.xml`.
Other configuration files, such as `web.xml` and `webcenter-config.xml` are described here too.

Topics:

- Configuration Files
- Cluster Configuration
- Configuration Tools
- Modifying the File Upload Size in Content Manager

See Troubleshooting Oracle WebCenter Portal Configuration Issues.

Configuration Files

`adf-config.xml`, `connections.xml`, and `web.xml` are used to configure WebCenter Portal and its back-end services. 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

adf-config.xml and connections.xml

`adf-config.xml` and `connections.xml` both store design time configuration information, such as the 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 mail server the application is currently using.
- **connections.xml** - Stores connection details for WebCenter Portal services.

  See *Developing Fusion Web Applications with Oracle Application Development Framework*.

After you deploy WebCenter Portal 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 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 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

For more information on MDS customizations, see Understanding the MDS Repository in *Administering Oracle Fusion Middleware*.

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`.

> **Note:**
>
> For detailed syntax and examples, see exportMetadata in *WLST Command Reference for Infrastructure Components*.

For example, to determine MDS customizations for `connections.xml` in WebCenter Portal, which has the application name `webcenter` and is deployed to the `WC_Portal` 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_Portal',
 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_Portal',
 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 on to Fusion Middleware Control and navigate to the home page for WebCenter Portal.
2. From the **WebCenter Portal** menu, select **System MBean Browser**.
3. Expand **Application Defined MBeans**.
4. Navigate to the MBean associated with the file you want to export.

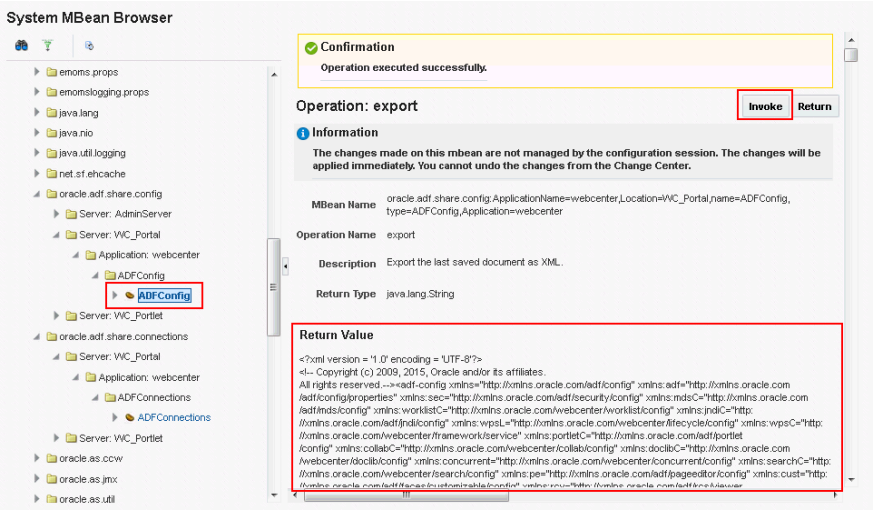
   For example, navigate to MBeans for `adf-config.xml` or `connections.xml` as follows:

   - `adf-config.xml` - **oracle.adf.share.config** > **Server: `WC_Portal`**> **Application: webcenter**> **ADFConfig**> **ADFConfig**
   - `connections.xml` - Click **oracle.adf.share.connections** > **Server: `WC_Portal`**> **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 B-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 application's configuration appears corrupt.

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_Portal',
      toLocation='/tmp/mydata',
      docs='/META-INF/mdssys/cust/adfshare/adfshare/adf-config.xml.xml')

     exportMetadata(application='webcenter', server='WC_Portal',
      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_Portal',
      fromLocation='/tmp/mydata',
      docs='/META-INF/mdssys/cust/adfshare/adfshare/connections.xml.xml')

     importMetadata(application='webcenter', server='WC_Portal',
      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.

  For detailed steps, see "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 exportMetadata and importMetadata in *WLST Command Reference for Infrastructure Components*.

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. For detailed syntax and examples, see exportMetadata in *WLST Command Reference for Infrastructure Components*.

1. Use the `exportMetadata` command to backup `connections.xml` and `adf-config.xml`.

   For example, for WebCenter Portal specify:

   ```
   exportMetadata(application='webcenter', server='WC_Portal',
    toLocation='/tmp/mydata',
    docs='/META-INF/mdssys/cust/adfshare/adfshare/connections.xml.xml')

   exportMetadata(application='webcenter', server='WC_Portal',
    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_Portal',
    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_Portal', 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.

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 `.par` 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.

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 *post deployment*. Oracle does not recommend that you explode application `.par` 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 WebCenter Portal, use the `uploadedFileMaxDiskSpace` parameter in `webcenter-config.xml` to configure a maximum upload size for files. For details, see webcenter-config.xml.

- **Time after which HTTP sessions expire**.

  See Specifying Session Timeout Settings.

- **JSP page timeout value**.

- **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. 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**.

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 Exploring the Settings Pages in WebCenter Portal Administration and Configuring Global Defaults Across Portals .

There are a few instances where you might be required to manually modify settings in `webcenter-config.xml`:

- **Maximum file upload size** (`uploadedFileMaxDiskSpace`) - the default setting is 2 GB. This setting is applicable when specifying the maximum upload size for files uploaded from features such as a wiki, blog, or activity stream.

> **Note:**
>
> For information about specifying the maximum upload size for files uploaded using Content Manager, see Modifying the File Upload Size in Content Manager.

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_Portal',
 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 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</
```

```
uploadedFileMaxDiskSpace>
</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_Portal',
 fromLocation='/tmp/mydata',
docs='/oracle/webcenter/webcenterapp/metadata/mdssys/cust/site/webcenter/
webcen
ter-config.xml.xml')
```

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 Starting and Stopping Managed Servers for WebCenter Portal Application Deployments.

In WebCenter Portal, 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, 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.

Configuration Tools

Oracle offers a range of tools for configuring WebCenter Portal 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 *Understanding 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 application's deployment plan. Typical properties changed on deployment include:

- Host/port properties for connections
- Standard J2EE artifacts in `web.xml`

> **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.

Modifying the File Upload Size in Content Manager

You can configure the maximum size for files uploaded through Content Manager. The default upload size is 50 MB.

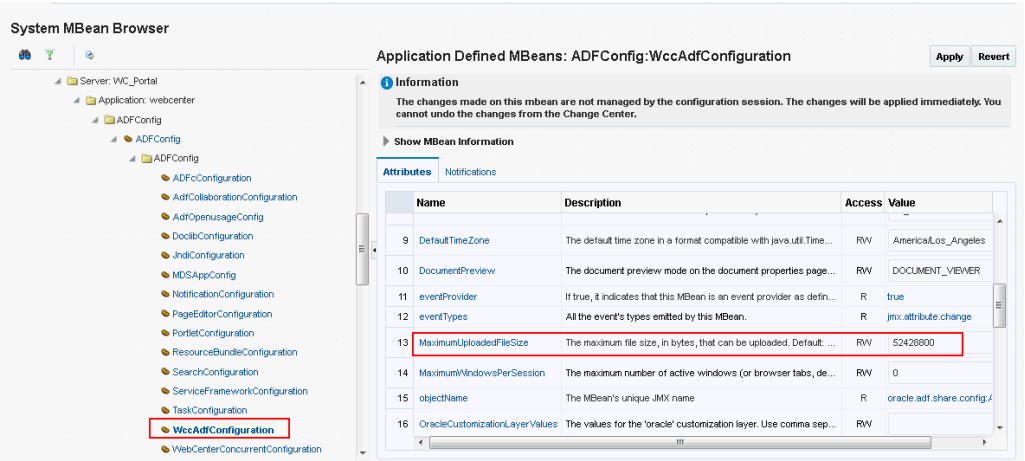
> **Note:**
>
> For information about specifying the maximum upload size for files uploaded from features such wiki, blog, or activity stream, see webcenter-config.xml.

To specify the maximum upload size allowed for files in Content Manager by using System MBean Browser:

1. Log on to Fusion Middleware Control and navigate to the home page for WebCenter Portal.
2. From the **WebCenter Portal** menu, select **System MBean Browser**.
3. Under **Application Defined MBeans**, under the `adf-config` MBean, navigate to the `WccAdfConfiguration` attribute:

   oracle.adf.share.config > Server: `WC_Portal` > Application: webcenter > ADFConfig > ADFConfig >ADFConfig > WccAdfConfiguration
4. In the `MaximumUploadedFileSize` attribute, specify the required file size in bytes.

Figure B-2 Modifying the MaximumUploadedFileSize Attribute

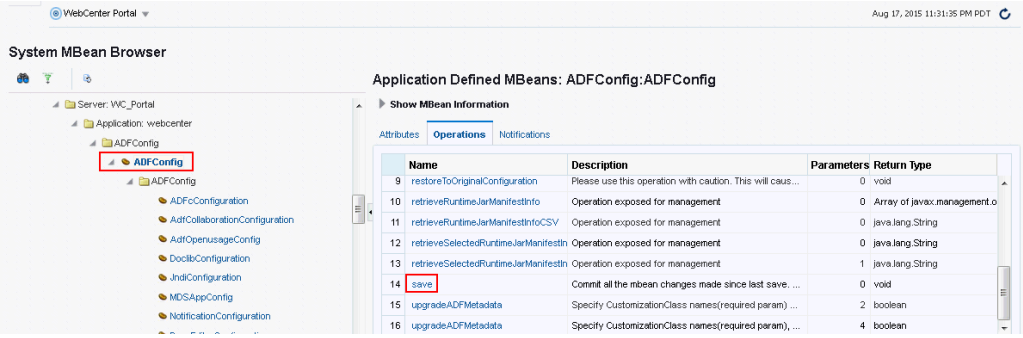


5. For a High Availability environment, you need to update the **Temporary Directory** attribute to specify the temporary location where files are stored. The **Temporary Directory** attribute must be set to a directory so that the uploaded files stored under that directory can be accessed by both node1 and node2.
6. Navigate to the adf-config MBean to invoke the save operation.

   Click **oracle.adf.share.config > Server: `WC_Portal` > Application: webcenter > ADFConfig > ADFConfig**
7. Click the **Operations** tab.

8. Click the **save** operation link.

Figure B-3 Saving the MBean Changes



9. On the Operation:save page, click **Invoke** to commit all the MBean changes made since the last save operation.
10. Restart `WC_Portal`, the WebCenter Portal managed server.

C

Third-Party Product Support

Use third party products with WebCenter Portal.
The following table lists the third party products that can be used with WebCenter Portal.

Table C-1 WebCenter Portal - Third Party Product Support

Feature	Product and Version	More information
Database	Microsoft SQL Server 2005 Microsoft SQL Server 2008	*Oracle Fusion Middleware Supported System Configurations*
Identity Store	Supported LDAP Identity Store Types	Default Identity and Policy Stores
Events	Microsoft Exchange Server 2007, Microsoft Exchange Server 2010, Microsoft Exchange Server 2013	Events Prerequisites for Personal Events
Mail	Microsoft Exchange Server 2007, Microsoft Exchange Server 2010, Microsoft Exchange Server 2013	Configuring Microsoft Exchange Server 2007, 2010, or 2013 for WebCenter Portal
Presence	Microsoft Lync 2010	Instant Messaging and Presence Server Prerequisites
Search	Elasticsearch	Understanding Search with Elasticsearch

Note:

Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

D

Migrating Wiki Content to WebCenter Portal

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.

> **Note:**
>
> Do not use the Document Migration Utility to export or import portal folders and portal template folders.
>
> To be able to perform the tasks listed in this appendix, 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.

Topics:

- Understanding Wiki Documents and Wiki Pages
- Migrating Data from the Source Wiki Application to WebCenter Portal

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 Working with Wikis in *Using Portals in Oracle WebCenter Portal*.

This section contains the following topics:

- Understanding Wiki Documents
- Understanding Wiki Pages

Understanding Wiki Documents

In WebCenter Portal you can create a wiki document within a wiki page. 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.

Understanding Wiki Pages

In WebCenter Portal you can create a wiki page by creating a page based on the `Wiki` page style. When you navigate to a wiki page you are presented with a wiki document. See Working with Wikis in *Using Portals in 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 wiki 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:

  `/RootFolder/`*`PortalFolder`*`/`*`wikiPageNameFolder`*`/`

  When FrameworkFolders is enabled, the wiki folder is created at the following path: /Enterprise Libraries/*`PortalFolder`*`/`*`wikiPageNameFolder`*`/`

- 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. Consider that 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: `/Enterprise Libraries/Marketing/Wiki1`
- Document: `/Enterprise Libraries/Marketing/Wiki1/Wiki1.htm`

  Attributes set for the document:
  - `dDocTitle= Wiki1.htm`
  - `dOriginalName = Wiki1.htm`
  - `dDocFunction = 'wiki'`
  - `dDocType = 'Application'`
  - `xWCPageID = Wiki1.jspx`

When you navigate to a wiki page the following occurs:

- Content Server is queried for the document in the following location:

  `/RootFolder/`*`PortalFolder/wikiPageNameFolder/wikiPageName.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.

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:

- Preparing WebCenter Portal for Importing Wiki Content
- Writing and Running a Custom Wiki Extraction Tool to Extract Content from the Wiki Application
- Using the Document Migration Utility to Import the Archive into the Target Portal
- Creating Wiki Pages in WebCenter Portal for the Content in WebCenter Content Server

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` in a Folders_g setup. For example:

```
<idc:field name="dCollectionGUID">05573322-E895-
EDA3-8A83-07CF39CBDE05</idc:field>
```

When using the FrameworkFolders folder service, search for the string `fApplicationGUID`, for example:

```
<idc:field name="fApplicationGUID">WC01:8c1c6442-a258-4cd7-8cf1-
adb60fc45ce2</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.

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.

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:

- Extracting and Arranging the Wiki Content
- Cleaning Up the Source HTML of Wiki Documents
- Rewriting the URLs
- Building the ExportImportData.xml Documents
- Building the Archive File

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 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 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)
```

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>
```

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 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/dDocName%3aFILE_ID"
     ```

   - Example:

     – Source URL:

       ```
       <img alt="image.jpg" src="images/image.jpg">
       ```

     – WebCenter URL:

       ```
       <img alt="image.jpg" resourceid="dev-ucm#dDocName:WSIMPORT4"
       src="http://webcenter.example.com/webcenter/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/webcenter/faces/
       owResource.jspx?
       z=oracle.webcenter.doclib%21sd15f13b8_141d_421b_ad0e_bc54b6f16893%21d
       ev-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/ProductBranding.pptx">
    Download Product Branding Presentation</a>
    ```

  - WebCenter URL:

    ```
    <a href="http://webcenter.example.com/webcenter/content/dev-
    ucm/uuid/dDocName%3aWSIMPORT7"> Download Product Branding
    Presentation</a>
    ```

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.

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 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.

The XSD for the ExportImportData.xml document is shown below.

```
<?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.

The following annotated example shows a partially complete `ExportImportData.xml` document, when FrameworkFolders is used as the folder service. Note that the example contains blank lines and XML comments that should not exist in a real `ExportImportData.xml` document.

```
<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="fFolderName"/>
      </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-->
      <!-- 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 value="0" name="fInhibitPropagation"/>
      </attributes>
    </document>
    <!-- closing tag for document 'Doc1' -->
  </documents>
  <!-- closing tag for all the documents in the portal root -->
</groupspace-folder>
```

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
```

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.

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.

Log into WebCenter Portal and navigate to the portals to which content was imported and ensure the content exists.

Properties Required to Run the Document Migration Utility

Table D-1 describes the properties required to run the Document Migration Utility. For information on how to run the utility, see Migrating Content Using the Document Migration Utility.

Table D-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
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

Table D-1 (Cont.) Document Migration Properties

Property	Description	Requirement
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
UCMSpacesRoot	Root folder under which WebCenter Portal content is stored. The value may be set as `/foldername`.	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

Migrating Content Using the Document Migration Utility

You can use any of the following methods to migrate content using the Document Migration Utility:

- 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 D-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
#   UCMSpacesRoot - Root folder where WebCenter Portal content is stored.
    Format: /foldername
# 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>
```

```
#UCMSpacesRoot=/portalrootfolder

# 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:

```
java -jar content-migration-tool.jar
<absolute_path_to_migrationPropertiesFilename>
```

For example:

```
java -jar content-migration-tool.jar /home/user1/
myMigrationProperties.properties
```

Optionally, specify logging settings using the `java.util.logging.config.file` parameter as described in Running the Document Migration Utility with Additional Logging.

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] UCMSpacesRoot
   ```

   To import content:

   ```
   java -jar content-migration.jar Usage UCMConn UCMUser TmpDirPath ArchvePath
   ArchiveName [UCMPwd] UCMSpacesRoot
   ```

   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 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 Running the Document Migration Utility with Additional Logging.

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
```

Creating Wiki Pages in WebCenter Portal for the Content in WebCenter Content Server

To use WebCenter Portal wiki pages to display the imported wikis, perform the following steps:

1. Log into WebCenter Portal.
2. Locate the portal where the content has been uploaded.
3. Select **Create Page** from the **Pages and Portals Actions** menu.
4. Select the **Wiki** page style.
5. In the **Title** field, enter a name for the wiki document, and click **Create**.

   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`.

   For more information about wiki pages, see Enabling Wikis in a Portal in *Building Portals with Oracle WebCenter Portal*.

E

Migrating Folders_g to FrameworkFolders

Migrate WebCenter Portal content from the Folders_g folder service to the FrameworkFolders folder service on Oracle WebCenter Content Server.

Topics:

- Understanding Folders_g Migration to FrameworkFolders
- Understanding the Folders_g and FrameworkFolders Directory Structure
- Migrating WebCenter Portal Data
- Troubleshooting Migration Issues

Understanding Folders_g Migration to FrameworkFolders

Oracle WebCenter Content offers two folder solutions: Folders_g and FrameworkFolders. The Folders_g component (aka the *Contribution Folders* interface) provides a hierarchical folder interface to content on Content Server. The FrameworkFolders component (aka 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.

Oracle WebCenter Portal supports the FrameworkFolders component on Content Server. For an Oracle WebCenter Portal instance patched from an earlier release that used the Folders_g folder service, you must migrate to the FrameworkFolders folder service.

> **Note:**
>
> FrameworkFolders is the name of the component that replaces the older Folders_g component. The older Folder interface is now referred to as Contribution Folders. The interface supported by the FrameworkFolders component is referred to as Folders.
>
> In this chapter, the migration process has been described using the terms FrameworkFolders and Folders_g. Unless referring to a UI selection or a command, the term FrameworkFolders can be used interchangeably with Folders, and the term Folders_g can be used interchangeably with Contribution Folders.
>
> For more information about the migration procedure, see Migrating Folders_g to Folders in *Administering Oracle WebCenter Content*.

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.

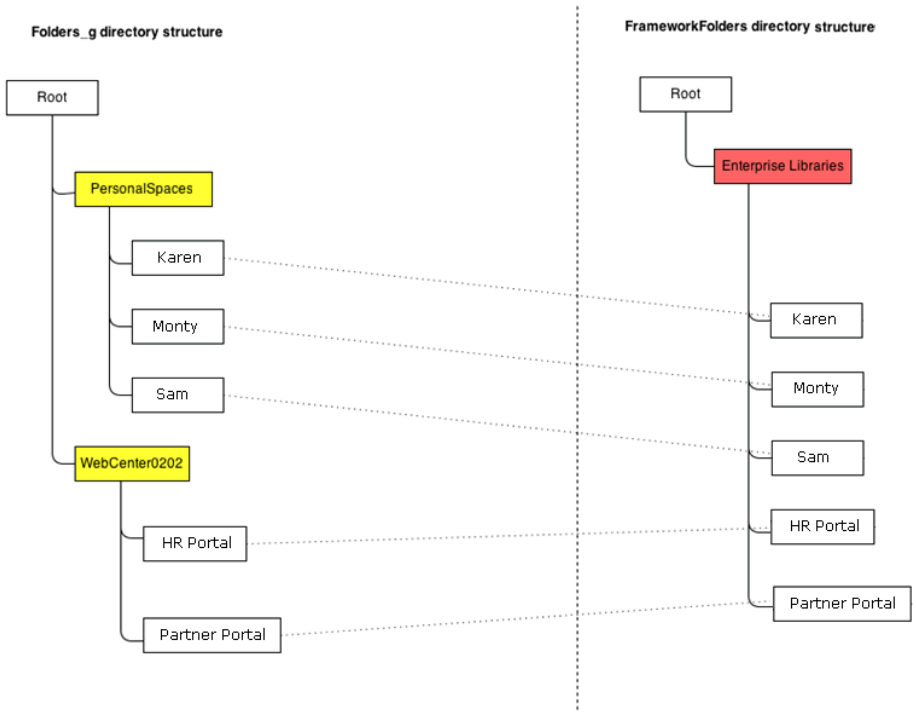
When WebCenter Portal uses a Content Server repository with Folders_g enabled, portals are stored under the WebCenter Portal root folder. Each user has a personal folder (the Home portal) stored under the path `/PersonalSpaces`, and this folder is named after the user. Whereas, when WebCenter Portal 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`

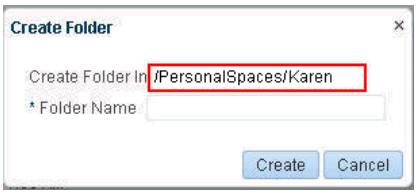
Figure E-1 shows how WebCenter Portal folders are organized in the Folders_g and FrameworkFolders setups on Content Server. In a Folders_g setup, personal folders for users `Karen`, `Monty`, and `Sam` are organized under `PersonalSpaces`, and the portals `HR Portal` and `Partner Portal` are organized 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 E-1 Folders_g and FrameworkFolders Folder Structure on Content Server



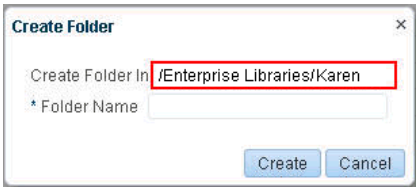
For example, if Karen creates any new folders in the Home portal in a Folders_g setup, the new folders are created under `/PersonalSpaces/Karen` as shown in Figure E-2.

Figure E-2 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 E-3.

Figure E-3 Folder Path When FrameworkFolders is Enabled



Item Level Security in Folders_g and FrameworkFolders

In the Folders_g setup, Item Level Security (ILS) can be set on files and folders. ILS settings on folders are inherited by all files within a folder unless a file has its own ILS defined. In the FrameworkFolders setup, ILS can be set only on files. ILS cannot be set on folders.

After you migrate from Folders_g to FrameworkFolders, you will not be able set ILS for a folder, but the ILS settings on each file contained in that folder are retained. You can update or remove file-level ILS settings after migration.

Migrating WebCenter Portal Data

This section describes the procedure for migrating WebCenter Portal content from Folders_g to FrameworkFolders.

This section includes the following subsections:

- Migration Roadmap
- Running exportFoldersGData to Generate the Pre-Migration Data
- Migrating WebCenter Portal MetaData to FrameworkFolders
- Running migrateFoldersGDataToFrameworkFolders to Validate the Migrated Data

Migration Roadmap

The flow chart (Figure E-4) and the table (Table E-1) in this section provide an overview of the steps required to migrate WebCenter Portal content from Folders_g to FrameworkFolders.

> **Note:**
>
> In a clustered environment, you must bring down all the managed servers except one WebCenter Portal managed server and one WebCenter Content managed server. Ensure that you run the migration WLST commands from the machine where WebCenter Portal is running.
>
> In a non-clustered environment, if WebCenter Portal and WebCenter Content managed servers run on different machines, run the migration WLST commands from the machine where WebCenter Portal is running.

Figure E-4 Migrating WebCenter Portal Data from Folders_g to FrameworkFolders

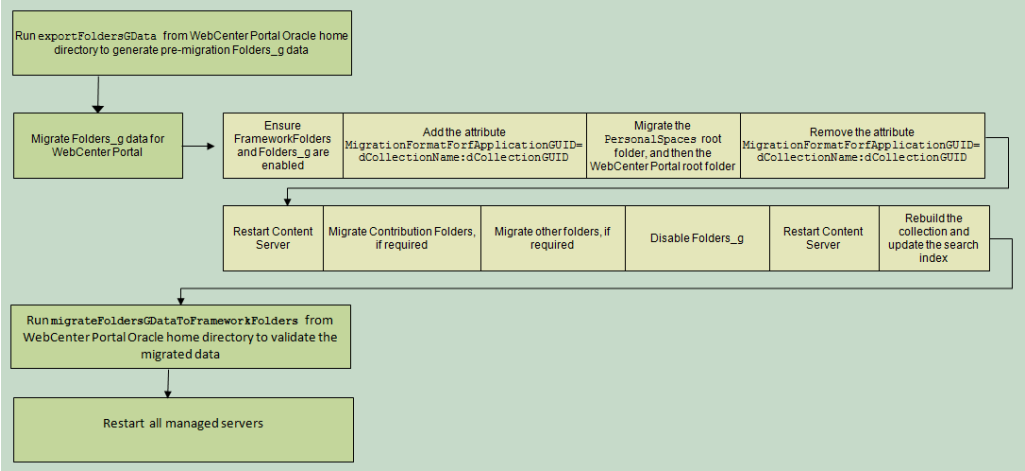


Table E-1 Migrating WebCenter Portal Data from Folders_g to FrameworkFolders

Task	Description	Documentation
Run the `exportFoldersGData` WLST command	Run the `exportFoldersGData` WLST command from the WebCenter Portal Oracle home directory to generate the pre-migration data.	Running exportFoldersGData to Generate the Pre-Migration Data.
Migrate Folders_g metadata for WebCenter Portal to FrameworkFolders	Run the migration utility on Content Server to migrate WebCenter Portal data from Folders_g to FrameworkFolders.	Migrating WebCenter Portal MetaData to FrameworkFolders.
Run the `migrateFoldersGDataToFrameworkFolders` WLST command	Run the `migrateFoldersGDataToFrameworkFolders` WLST command from the WebCenter Portal Oracle home directory to validate the migrated data.	Running migrateFoldersGDataToFrameworkFolders to Validate the Migrated Data.

Table E-1 (Cont.) Migrating WebCenter Portal Data from Folders_g to FrameworkFolders

Task	Description	Documentation
Restart all the managed servers	Restart all the managed servers, including the WebCenter Portal and Content Server managed servers.	See Starting and Stopping Oracle WebLogic Server Instances in *Administering Oracle Fusion Middleware*.

Running exportFoldersGData to Generate the Pre-Migration Data

Use the WLST command `exportFoldersGData` to generate the Folders_g pre-migration data:

```
exportFoldersGData(appName,server,[connectionName,directoryPath,applicationVersion])
```

The following example exports the Folders_g data for a WebCenter Portal application deployed to the `WC_Spaces` managed server. The content server connection named `MyContentServerConnection` is used, and Folders_g data exported to the `/scratch/myTemp_Dir` directory.

```
exportFoldersGData(appName='webcenter',server='WC_Spaces',connectionName='MyContentServerConnection',directoryPath='/scratch/myTemp_Dir/')
```

For command syntax and examples, see exportFoldersGData in *WebCenter WLST Command Reference Reference*. You must run this command from the WebCenter Portal Oracle home directory. For information about how to run WLST commands, see Running Oracle WebLogic Scripting Tool (WLST) Commands.

The pre-migration data generated by `exportFoldersGData` can be used to generate data consistency reports and ascertain the sanity of the migrated files and folders after you have migrated WebCenter Portal data to FrameworkFolders.

The `exportFoldersGData` WLST command performs the following tasks:

- Generates the pre-migration metadata for all portal folders under the WebCenter Portal root folder
- Generates the pre-migration metadata for all user folders under the Home portal root folder
- Writes the generated metadata to the `PreMigrationData.csv` file.
- Exports all MDS documents

The `PreMigrationData.csv` file includes the following metadata for the portal folders and user folders stored in the Folders_g data structure:

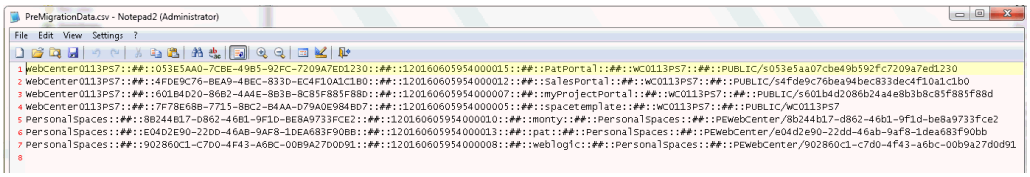
- Folder name for the WebCenter Portal root folder or the Home portal root folder
- `dCollectionGUID`, the identifier used by WebCenter Portal to map a portal to the corresponding folder on Content Server
- `dCollectionID`, the identifier used in Folders_g to uniquely identify a folder
- `dCollectionName`, the folder name

- `dSecurityGroup`, the security group
- `dDocAccount`, the account name

By default, `PreMigrationData.csv` is stored at the following path: *`WCP_ORACLE_HOME`*`/common/wlst/FG_FF_MIGRATION`. You can choose a different location for the file while running the `exportFoldersGData` WLST command.

Figure E-5 shows a sample `PreMigrationData.csv` file. Each row displays the metadata for a portal folder or a user folder.

Figure E-5 Sample PreMigrationData.csv File



Migrating WebCenter Portal MetaData to FrameworkFolders

You use the Folders Migration utility available on Content Server to migrate WebCenter Portal metadata and folder structure from Folders_g to FrameworkFolders.

To migrate WebCenter Portal metadata from Folders_g to FrameworkFolders:

1. Log on to Content Server as an administrator.
2. Enable Framework Folders. Ensure Folders_g is also enabled. For information, see Enabling Mandatory Components.

> **Note:**
>
> When you migrate from the Folders_g to the FrameworkFolders setup, both the Folders_g and FrameworkFolders components must be enabled during the migration process.

3. Add the required attributes:
   a. Log on to Content Server.
   b. Navigate to **Administration > Admin Server > General Configuration**.
   c. In the **Additional Configuration Variables** box, add the following entry:

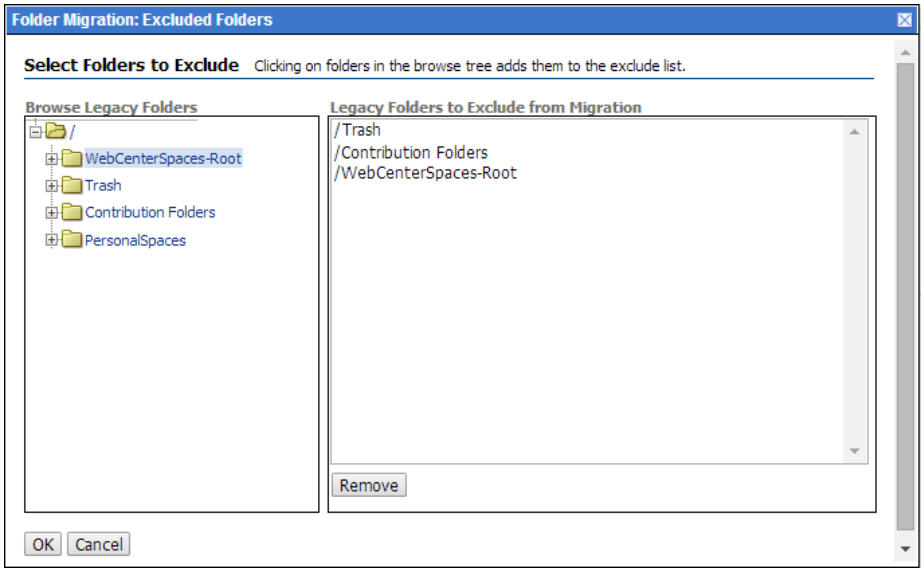
      ```
      MigrationFormatForfApplicationGUID=dCollectionName:dCollectionGUID

      DisableQueryTimeoutSupport=true
      ```

   d. Click **Save**.
4. Restart Content Server, and log on as an administrator.
5. Choose **Administration**, then **Folder Migration** to begin performing data migration.
6. Migrate the **PersonalSpaces** root folder.

a. Under the Run Migration section on the Folder Migration page, click **Modify Excluded Folders**.

b. In the Folder Migration: Excluded Folders dialog, specify the folders that need to be excluded from migration. Ensure that all folders other than the **PersonalSpaces** root folder appear in the **Legacy Folders to Exclude from Migration** box. In Figure E-6, `WebcenterSpaces-Root` refers to the WebCenter Portal root folder.

Figure E-6 Specifying Folders Excluded from Migration



c. Click **OK**.

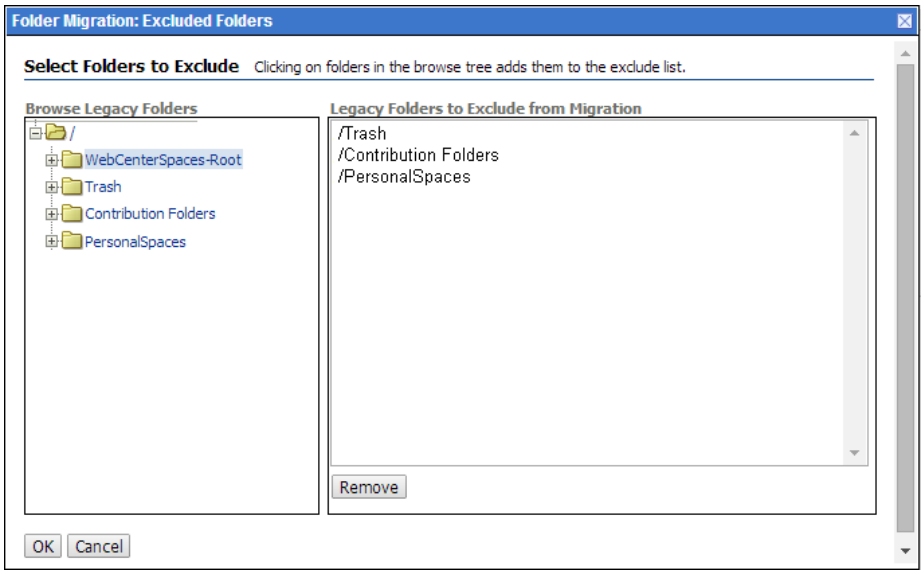
d. Click **Migrate Folder Data** to migrate the data.

7. Migrate the WebCenter Portal root folder.

a. Under the Run Migration section, click **Modify Excluded Folders**.

b. In the Folder Migration: Excluded Folders dialog, ensure that all folders other than the WebCenter Portal root folder appear in the **Legacy Folders to Exclude from Migration** box. In Figure E-7, `WebcenterSpaces-Root` refers to the WebCenter Portal root folder.

Figure E-7 Specifying Folders Excluded from Migration



c. Click **OK**.

d. Click **Migrate Folder Data** to migrate the data.

8. Remove the `MigrationFormatForfApplicationGUID` attribute added in step 3.

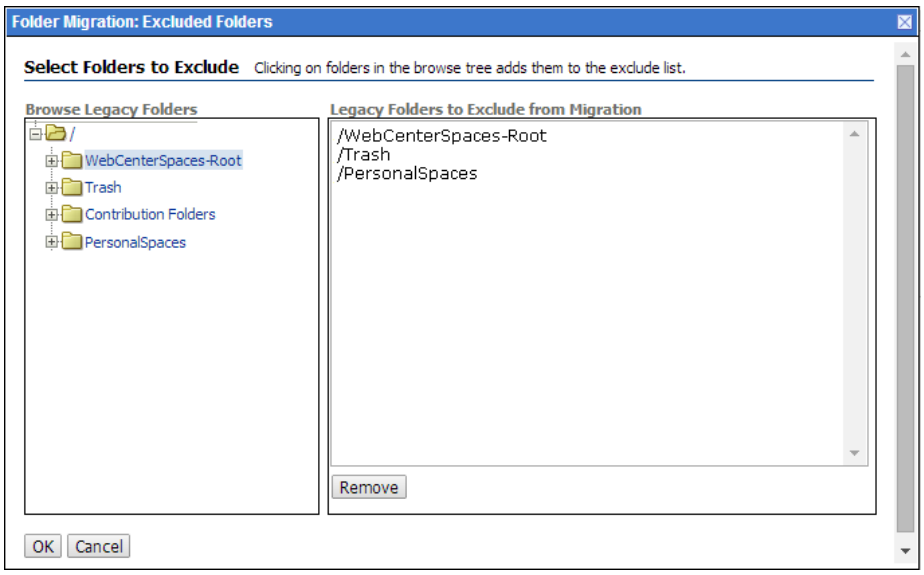
9. Restart Content Server.

10. If Contribution Folders contains any content that is referenced in WebCenter Portal, you must migrate the Contribution Folders folder as an enterprise library.

    a. In the Run Migration section, click **Modify Excluded Folders**.

    b. In the Folder Migration: Excluded Folders dialog, ensure that all folders other than **Contribution Folders** appear in the **Legacy Folders to Exclude from Migration** box (Figure E-8).

Figure E-8 Specifying Folders Excluded from Migration



c. Click **OK**.

d. In the Folder Migration Destination section, click **Browse** to specify the destination for the folders to be migrated.

e. In the Browse dialog, select **Enterprise Libraries**, and click **OK**.

f. Click **Migrate Folder Data**.

11. Migrate folders other than PersonalSpaces, the WebCenter Portal root folder, and Contribution Folders if they contain any content that is referenced in WebCenter Portal. Follow the same procedure that you used in step 10.

    In the Folder Migration: Excluded Folders dialog, ensure that all folders other than the folder you want to migrate are listed in the **Legacy Folders to Exclude from Migration** box.

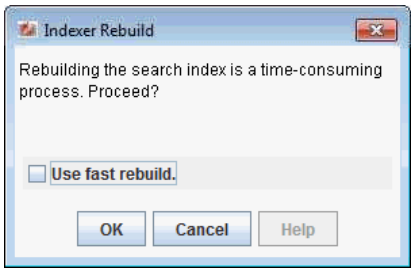
12. Disable Folders_g.

    On the Advanced Component Manager page, from the **Enabled Components** list box, select **Folders_g** and click **Disable**.

13. Restart Content Server.

14. Rebuild the collection and update the search index using the Repository Manager utility. When rebuilding the collection, ensure that the **Use fast rebuild** check box is unchecked in the Indexer Rebuild dialog (Figure E-9). For information, see Working with the Search Index in *Administering Oracle WebCenter Content*.

Figure E-9 Rebuilding the Index



Logs generated during the migration process are available at *WCC_DOMAIN*`/servers/UCM_server1/logs`, where *WCC_DOMAIN* refers to the Oracle WebCenter Content domain.

Running migrateFoldersGDataToFrameworkFolders to Validate the Migrated Data

Use the `migrateFoldersGDataToFrameworkFolders` WLST command to migrate Folders_g data to FrameworkFolders and check the integrity of the migrated data:

```
migrateFoldersGDataToFrameworkFolders(appName,server,contentDbConnectionUrl,contentDbUserName,[connectionName,directoryPath,reportMode,applicationVersion])
```

The following example migrates Folders_g data from the `/scratch/myTemp_Dir` directory to FrameworkFolders and validates the migrated data for the WebCenter Portal application deployed to the `WC_Portal` managed server. Content Server connection named `MyContentServerConnection` and the specified WebCenter Content database connection and username are used to perform the migration.

```
migrateFoldersGDataToFrameworkFolders(appName='webcenter',server='WC_Portal',contentDbConnectionUrl='wccdbhost.example.com:wccdbport:wccdbsid',
```

```
contentDbUserName='SCHEMA_PREFIX_OCS',connectionName='MyContentServerConnection',
directoryPath='/scratch/myTemp_Dir/)
```

The path specified in the `directoryPath` attribute must be the same that you specified while running the `exportFoldersGData` WLST command.

For command syntax and examples, see exportFoldersGData in *WebCenter WLST Command Reference Reference*. For information about how to run WLST commands, see Running Oracle WebLogic Scripting Tool (WLST) Commands.

The `migrateFoldersGDataToFrameworkFolders` command performs the following tasks:

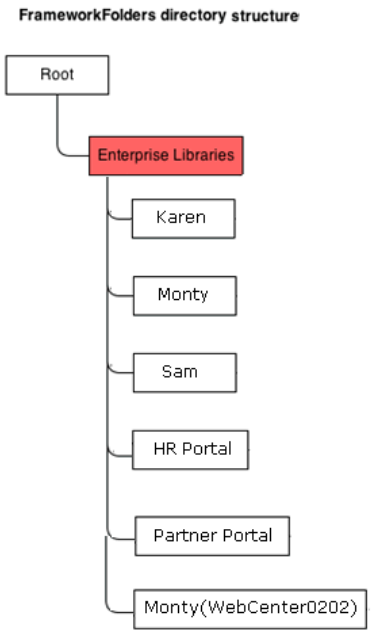
- Generates the metadata for the migrated WebCenter Portal content and validates it against the pre-migration metadata generated by the `exportFoldersGData` WLST command. It verifies that each migrated folder is an enterprise library, and the values for the account name and the security group are same as the pre-migration metadata.

  Any data inconsistency is reported in the summary text displayed after the command is executed, and is also stored in a log file available at this default location: *`WCP_ORACLE_HOME`*`/common/wlst/POST_MIGRATION/MigrationDiagnostic.log`.

- Creates `MigrationMap.csv`, a mapping file that maps the Folders_g identifier `dCollectionID` to the FrameworkFolders identifier `fFolderGUID`. The mapping file is stored here: *`Migration_Directory`*`/POST_MIgration/MigrationMap.csv`.

  In case of any folder name changes, the file also contains a mapping of the Folders_g path and the FrameworkFolders path. For example, suppose there is a user folder named `Monty`, and also a portal folder by the same name. In the Folders_g setup, the user folder `Monty` appears under the PersonalSpaces root folder, and the portal folder `Monty` appears under the WebCenter Portal root folder (such as, `WebCenter0202`). During migration, PersonalSpaces folders are migrated first. To resolve the folder name conflict, the portal folder is renamed as `Monty(WebCenter0202)`, as shown in Figure E-10. This folder path change is stored in the mapping file.

Figure E-10 FrameworkFolders Directory Structure After Migration



- Using the mapping file, replaces the `dCollectionID` value with the `fFolderGUID` value in the MDS documents generated by the `exportFoldersGData` WLST command. It also replaces any old path references containing the WebCenter Portal root folder name or PersonalSpaces, with the new path. For example, `/PersonalSpaces/weblogic` is replaced with `/Enterprise Libraries/weblogic`.

  Any inconsistencies are reported in `MigrationDiagnostic.log`. For troubleshooting information, see Troubleshooting Migration Issues.
- Imports all the updated MDS documents.

Troubleshooting Migration Issues

This section provides information to assist you in troubleshooting the problems you may encounter while migrating the Folders_g setup to the FrameworkFolders setup.

Problem

The migration summary displayed for the `migrateFoldersGToFrameworkFolders` WLST command contains a warning that a few MDS documents contain Folders_g path references that do not adhere to any known patterns.

Solution

Manually verify the `MigrationDiagnostic.log` file present in the *`WCP_ORACLE_HOME`*`/common/wlst/POST_MIGRATION` directory and look for the warning messages. For the specified files, verify whether the Content Server path reference is valid. If required, manually update the Folders_g path to the FrameworkFolders path and import the file.

Problem

When you run the `exportFoldersGData` WLST command for a non-primary Content Server connection for WebCenter Portal, the following exception is reported in the `MigrationDiagnostic.log` file:

```
oracle.stellent.ridc.protocol.ServiceException: Unable to open folder.
```

Solution

When you have multiple Content Server connections registered, to run the `exportFoldersGData` WLST command using a non-primary Content Server connection, set it as the active connection and specify the Root Folder and Application Name values. After running `exportFoldersGData`, set the active connection back to the previous Content Server connection and continue with migration.

Problem

Your WebCenter Portal application includes custom codes that use various Folders_g services-based queries. After migration to FrameworkFolders, the queries do not work.

Solution

Folders_g services cannot be used in WebCenter Portal migrated to FrameworkFolders. If your application includes Folders_g services, after migration you must replace them with the equivalent FrameworkFolders services. Table E-2 provides a mapping of Folders_g services and the corresponding FrameworkFolders services.

For information about FrameworkFolders services, see Folders Services in *Services Reference for Oracle WebCenter Content*.

Table E-2 Mapping of Folders_g Services with FrameworkFolders Services

Folders_g Service	FrameworkFolders Service
COLLECTION_ADD	FLD_CREATE_FOLDER
COLLECTION_BROWSE	FLD_INFO
COLLECTION_COPY_COLLECTION	FLD_COPY (with item1 specified as a folder)
COLLECTION_COPY_ITEM	FLD_COPY (with item1 specified as a file)
COLLECTION_COPY_LOT	FLD_COPY (with items specified)
COLLECTION_DELETE	FLD_DELETE
COLLECTION_DELETE_COLLECTION	FLD_DELETE (with item1 specified as a folder)
COLLECTION_DELETE_ITEM	FLD_DELETE (with item1 specified as a file)
COLLECTION_DELETE_LOT	FLD_DELETE (with items specified)
COLLECTION_DISPLAY	FLD_BROWSE
COLLECTION_GET_COLLECTIONS	FLD_RETRIEVE_CHILD_FOLDERS
COLLECTION_GET_CONTENTS	FLD_RETRIEVE_CHILD_FILES
COLLECTION_GET_LINKS	FLD_INFO (with item1 specified as a shortcut)
COLLECTION_GET_REFERENCE	FLD_INFO (with path specified or item1 specified as a path)
COLLECTION_INFO	FLD_INFO
COLLECTION_MOVE_ALL	FLD_MOVE (with items specified)
COLLECTION_MOVE_COLLECTION	FLD_MOVE (with item1 specified as a folder)
COLLECTION_MOVE_ITEM	FLD_MOVE (with item1 specified as a file)
COLLECTION_MOVE_LOT	FLD_MOVE (with items specified)

Table E-2 (Cont.) Mapping of Folders_g Services with FrameworkFolders Services

Folders_g Service	FrameworkFolders Service
COLLECTION_SEARCH_RESULTS	FLD_FOLDER_SEARCH
COLLECTION_UPDATE	FLD_EDIT_FOLDER

F

Troubleshooting WebCenter Portal

Discover tips for troubleshooting issues you may encounter while working with WebCenter Portal.

> **Note:**
>
> Oracle WebCenter Portal has deprecated the support for Jive features (announcements and discussions). If you have upgraded from a prior release to Release 12c (12.2.1.4.0), Jive features remain available in your upgraded instance but Oracle support is not provided for these features. In the next release, Jive features will not be available even in the upgraded instances

Topics:

- Using My Oracle Support for Additional Troubleshooting Information
- Troubleshooting Oracle WebCenter Portal Configuration Issues
- Troubleshooting Oracle WebCenter Portal WLST Command Issues
- Troubleshooting Oracle WebCenter Portal Performance Issues
- Troubleshooting WebCenter Portal Workflows
- Troubleshooting WebCenter Portal Import and Export
- Troubleshooting Individual Portal and Portal Template Import and Export
- Troubleshooting Issues with Mail
- Troubleshooting Issues with Announcements and Discussions
- Troubleshooting Issues with Events
- Troubleshooting Issues with Users and Roles
- Troubleshooting Issues with Content Repositories
- Troubleshooting Issues with Analytics
- Troubleshooting Issues with Notifications
- Troubleshooting External Application Issues
- Troubleshooting Security Configuration Issues
- Troubleshooting Issues with External Links
- Troubleshooting Issues with Elasticsearch

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`.

Troubleshooting Oracle WebCenter Portal Configuration Issues

This section includes the following subsections:

- Configuration Options Unavailable
- Logs Indicate Too Many Open Files

Configuration Options Unavailable

Problem

When you try to configure WebCenter Portal 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, Viewing and Configuring WebCenter Portal Logs.

Analyze messages for the modules `oracle.adf.mbean.share.connection` and `oracle.adf.mbean.share.config`, and determine what must be done.

Logs Indicate Too Many Open Files

Problem

WebCenter Portal 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.

Troubleshooting Oracle WebCenter Portal WLST Command Issues

This section includes the following topics:

- No Oracle WebCenter Portal WLST Commands Work
- Connection Name Specified Already Exists
- WLST Shell is Not Connected to the WebLogic Server
- More Than One Application with the Same Name Exists in the Domain
- More Than One Application with the Same Name Exists on a Managed Server
- Already in Domain Runtime Tree Message Displays

See also, Running Oracle WebLogic Scripting Tool (WLST) Commands.

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 **Oracle home directory** (*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, Running Oracle WebLogic Scripting Tool (WLST) Commands.

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. 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.

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, No Oracle WebCenter Portal WLST Commands Work and Running Oracle WebLogic Scripting Tool (WLST) Commands.

More Than One Application with the Same Name Exists in the Domain

Problem

You attempt to perform an operation on WebCenter Portal, 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 WebCenter Portal using the following WLST command:

```
registerWSRPProducer(appName='webcenter', 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='webcenter', name='MyWSRPSamples', url='http://
myhost.com:9999/portletapp/portlets/wsrp2?WSDL', server=WC_CustomPortal2)
```

See also, Running Oracle WebLogic Scripting Tool (WLST) Commands.

More Than One Application with the Same Name Exists on a Managed Server

Problem

You attempt to perform an operation on WebCenter Portal 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, Running Oracle WebLogic Scripting Tool (WLST) Commands.

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.

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 Identify Slow Pages
- How to Identify Slow Page Components
- How to Troubleshoot Slow Page Requests
- How to Troubleshooting Requests using JRockit Flight Recordings

About Performance Monitoring and Troubleshooting Tools

Various tools are available for monitoring and troubleshooting performance issues with your Oracle WebCenter Portal environment.

Table F-1 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 Cloud 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
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

How to Identify Slow Pages

Use Fusion Middleware Control to determine the slowest pages in WebCenter Portal. 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, as described in Navigating to the Home Page for WebCenter Portal.
2. From the **WebCenter Portal** menu, select **Monitoring** > **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, 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.

See also, Understanding Page Request Metrics and Customizing Key Performance Metric Thresholds and Collection.

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.

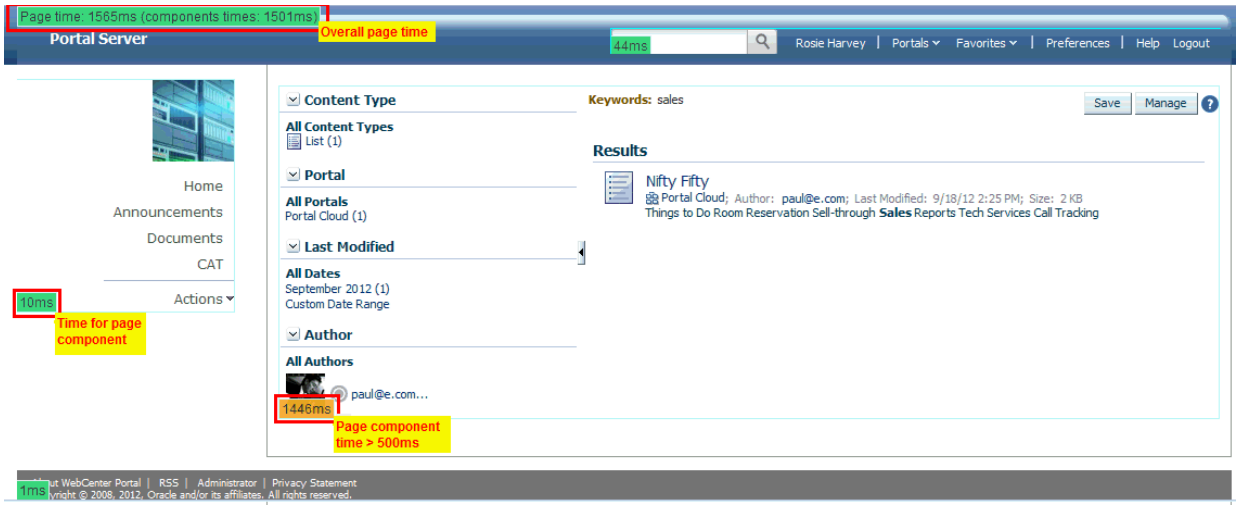
This section includes the following subsections:

- 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

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 F-1).

Figure F-1 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. For more consistent results, refresh the pages by clicking through the testing pages before enabling Page Performance Analysis.

Color Coding

Performance timings display in various colors to help alert you to problem areas. Refer to the following table:

Color	Time to Display
Green	< 100 ms
Green/Yellow	100 - 500 ms
Yellow	500 ms - 1 second
Orange	1 - 3 seconds
Red	> 3 seconds

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_Portal', 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_Portal', 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 Displaying and Hiding Page Timing Information for Your Current Session.

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 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 F-2).

   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 F-1).

   All subsequent pages that you display show timing information as well.

Figure F-2 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 F-2).

   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.

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 How to Identify Slow Pages.

2. Append `&perfDebug=on` to the page URL to display timing information for the page.

   See also, 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 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.

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

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, How to Troubleshooting Requests using JRockit Flight Recordings.

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.

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 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 Diagnosing Problems in the *Administering Oracle Fusion Middleware*.

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.
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, How to Troubleshooting Requests using JRockit Flight Recordings.

The ECID of the request that caused the stuck thread is recorded in the error message.

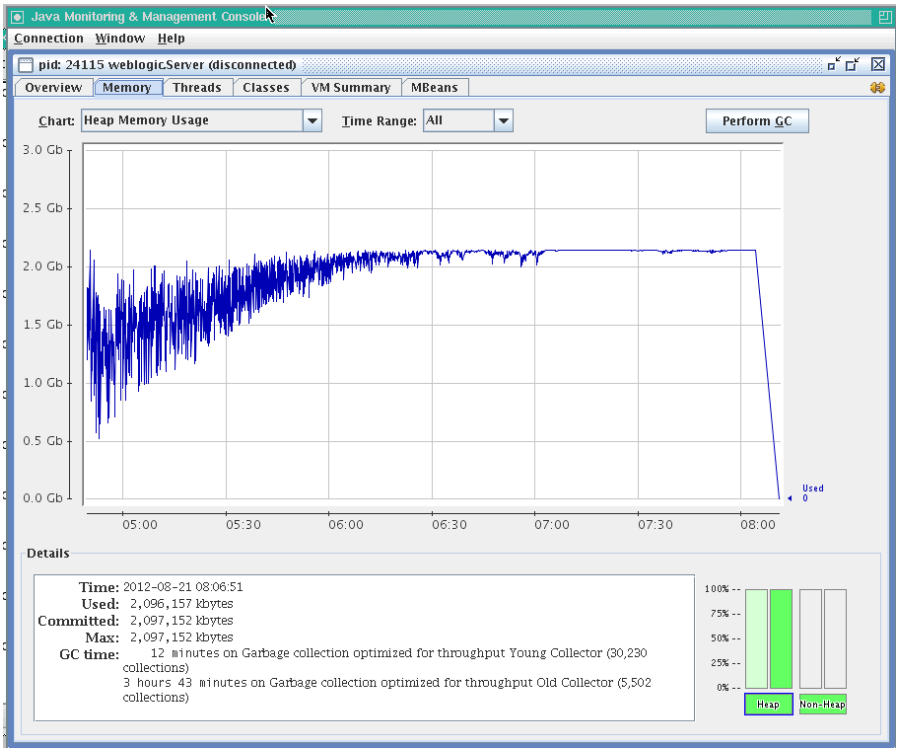
Troubleshooting Slow Requests Using JFR Recordings

See How to Troubleshooting Requests using JRockit Flight Recordings.

Troubleshooting Memory Leaks and Heap Usage Problems

If WebCenter Portal 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 F-3 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`.
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.

   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.

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 Viewing System Audit Information in *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 Server-Wide Tracing in *Administering Oracle WebCenter Content*.

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
   ```

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:
a. In the JRockitFlightRecorder.jfr page, click the **Events** icon. b. Click the **Log** tab at the bottom of the page. c. In the **Event Type** navigation pane on the left, locate **Dynamic Monitoring System** and then **HttpRequest**. d. Click **HTTP request**; de-select all the other event types. e. 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. f. Click the row in the table to view the attributes of the requests. g. In the **Event Attributes** sections, note the start time and the thread that serviced the request.	a. 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 Viewing and Searching Log Files in the *Administering Oracle Fusion Middleware.* b. 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. c. Click the **ECIDs** tab at the bottom of the page. d. In the **ECIDs** section, from **Filter Column** list, select **ECID**. e. Enter the ECID in the search box and select <Enter>. f. In the results table, highlight the row with the matching ECID and right-click to bring up the menu. g. 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. h. Click the **Events** icon. i. In the Event Type navigation pane on the left, locate **Dynamic Monitoring System** and then **HttpRequest**. j. Click **HTTP request**; de-select all the other event types. ** In the **Event Log** section, click **Show Only Operative Set**. Each row corresponds to the request with the matching ECID. k. Click the row in the table to view the attributes of the requests. l. 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 Analyzing Flight Recorder Data in JRockit Mission Control in *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.

Troubleshooting WebCenter Portal Workflows

If you experience issues with WebCenter Portal workflows, review the following sections:

- Validating the WebCenter Portal Workflow Configuration
- Troubleshooting Issues with WebCenter Portal Workflows
- Email Notifications Not Working

Email Notifications Not Working

Problem

Notifications for workflows are not being sent by email to BPM Worklist, as described in Configuring WebCenter Portal Workflow Notifications to be Sent by Email.

> **Note:**
>
> To identify causes of failures, examine log files on the managed SOA server, hosting BPM Worklist processes, you have configured.

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.

> **Note:**
>
> To identify causes of failures, examine log files for any SOA BPEL servers you have configured.

Validating the WebCenter Portal Workflow Configuration

The *Installing and Configuring Oracle WebCenter Portal* describes how to install and configure WebCenter Portal workflows. For details, see Back-End Requirements for WebCenter Portal Workflows . 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 to BPM Worklists as `User2`.

   You will be able to view the notification in your worklist items that you have been added to the portal in the specified role.
5. Open the invite notification and click the **Acknowledge** button.

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.

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 Oracle SOA Server - Extending the Domain in *Installing and Configuring 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
```

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 Creating the SOA Domain Keystore.

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 Overview of Oracle WebCenter Portal WLST Command Categories in *WebCenter WLST Command Reference Reference*.

Troubleshooting WebCenter Portal Import and Export

This section contains the following subsections:

- ResourceLimitException Issue
- LockRefreshTask Issue
- Portals and Portal Templates Not Available After Import

- Unable to Migrate Portals or Documents If the Source and Target Applications Share the Same Content Server

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.

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_Portal,name=Lif
ec
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_Portal', docs='/oracle/
webcenter/lock/applicationImport/applicationImport.xml')
```

For WebCenter Portal application export failure, run:

```
deleteMetadata(application='webcenter', server='WC_Portal', docs='/oracle/webcenter/
lock/applicationExport/applicationExport.xml')
```

For portal import failure, run:

```
deleteMetadata(application='webcenter', server='WC_Portal', docs='/oracle/webcenter/
lock/scopeImport/**')
```

For portal export failure, run:

```
deleteMetadata(application='webcenter', server='WC_Portal', docs='/oracle/webcenter/
lock/gsExportImport/**')
```

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

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.

Target Portal Server Shown As Unavailable When Creating a Connection

While creating a Portal Server connection to a target server, you can test the connection by clicking the Test button. After clicking Test, if you view the state of the target Portal Server in its administration console, it is shown as unavailable even if the server is up and running. This happens when the name of the target domain or server is same as the source domain or server. This does not lead to any functionality loss.

Troubleshooting Individual Portal and Portal Template Import and Export

This section contains the following subsections:

- Portal Blocked After Unsuccessful Export or Import
- Page or Portal Not Found Message After Import
- Portal Import Archive Exceeds Maximum Upload File Size
- Maximum Number of Portals Exceeded on Export
- Lists Not Imported Properly
- Exporting and Importing Portals with Tools and Services Configured
- Tools and Services Disabled After Import
- Importing from the Subportals Page
- Unable to Import a Portal If the Source and Target Applications Share the Same Content Server
- Shared Library Changes Not Available after Portal Deployment
- Members Not Listed in an Imported Portal
- Deployment Messages Not Displayed in the Browser Locale

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 Taking Any Portal Offline and Bringing Any Portal Back Online. See also, the WLST command setSpaceState in *WebCenter WLST Command Reference Reference*.

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.

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 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 Changing the Maximum File Upload Size.

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 *Tuning Performance*.

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.

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_28d487cb3bce/**"/>
               </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.

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.

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.

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.

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.

Shared Library Changes Not Available after Portal Deployment

Problem

Changes made to the shared library are not showing up in the newly deployed portal.

Solution

Shared library deployment might have failed. To find out about the deployment status of the shared library, go to the WebLogic Server Admin console on the target instance and check the state of the latest version of the shared library that you deployed. The newly deployed library must be in "active" state on the target.
If the shared library deployment failed, and the state of the last deployed version on the target is shown as failed, delete the failed version and either redeploy the portal or propagate the portal to include shared library changes.

If the shared library deployment failed, and the shared library on the target is not shown in the failed state but some other intermediate state (such as Distribute running/Deploy running/ New), try restarting the servers and then click on **Activate Changes** in the console (if it is enabled). If the shared library state is still not shown as active, delete the shared library version from the target server and redeploy or propagate the portal. If you get an error message that this version of the library cannot be deleted as it is being referenced by one or more applications, stop the WebCenter Portal managed server (`WC_Portal`), delete the library, and then start the managed server again. After restart, it will start using the last active version of the shared library.

Members Not Listed in an Imported Portal

Problem

After you have imported a portal from a portal archive (PAR file) containing members under various roles, like Viewer or Portal Manager, member names are not displayed on the Portal's members page.

Solution

Users on both the source and target instances must be identical. If a shared identity store is not used, your system administrator must synchronize users in WebCenter Portal by using the `synchronizeUserInformation` WLST command. For information about this command, contact Oracle Support.

Deployment Messages Not Displayed in the Browser Locale

When you deploy a portal, the deployment progress messages coming from the target server might be displayed in the source server’s environment locale instead of the browser locale. This happens when your source server's environment locale is different than the browser locale set in WebCenter Portal.

Troubleshooting Issues with Mail

This section includes the following subsections:

- Mail is Not Accessible in Secure Mode

- Mail is Not Accessible in Non-Secure Mode
- Unable to Create Distribution Lists in the Non-Secure Mode
- Unable to Create Distribution Lists in the Secure Mode
- Provisioning of Mail Fails in a Portal (Default Distribution List not Created)
- Unable to Configure the Number of Mail Messages Downloaded
- Unable to Publish and Archive WebCenter Portal Mail
- Changing Passwords on Microsoft Exchange

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 Registering Mail Servers.
- Properties are set to `true` in your mail server.
  - `mail.imap.secured = true`
  - `mail.smtp.secured = true`

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, Registering Mail Servers.
- Properties are set to `false` in your mail server.
  - `mail.imap.secured = false`
  - `mail.smtp.secured = false`

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 Registering Mail Servers.

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:**
>
> Registering Mail Servers

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.

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` in *WebCenter WLST Command Reference Reference*.

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 Publishing Portal Mail in a Discussion Forum in *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 Publishing Portal Mail in a Discussion Forum in *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.

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.

Troubleshooting Issues with Announcements and Discussions

This troubleshooting section includes the following subsections:

- Authentication Failed
- Discussions Cannot Be Enabled in WebCenter Portal
- Login Failed
- Login Does Not Function Properly After Configuring Oracle Access Manager
- Category Not Found Exceptions
- Watched Topics and Recent Topics Not Displaying Topics From Multiple Discussion Forums
- Discussion and Announcement Updates Not Displayed
- Announcements Page Displays "User Is Not Authorized"
- Discussions Page Displays "User Is Not Authorized"

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_Portal-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.

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 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 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 Specifying Where Discussions and Announcements are Stored on the Discussions Server .

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(JiveAuthentic
ator.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 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`.

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 addDiscussionsServerAdmin in *WebCenter WLST Command Reference Reference*.
2. Log on to the Discussions Admin Console with the super admin account, and navigate to System - System Properties.

   See 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 Configuring the Discussions Server for SSO.
4. Restart WebCenter Portal's Discussion Server.

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.

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.

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.

Announcements Page Displays "User Is Not Authorized"

Problem

A user creates an announcement, then changes their user name. When they try to log in with the new user name to access the page, a message indicating that the user is not authorized to view the announcement is displayed even if the two user names were synced.

Solution

This can occur if the same Identity Store is not used by WebCenter Portal and the Discussions server; the user must exist in both stores if the store is not shared.

Discussions Page Displays "User Is Not Authorized"

Problem

A user creates a discussion topic, then changes their user name. When they try to log in with the new user name to access the page, a message indicating that the user is not authorized to view the discussion page is displayed even if the two user names were synced..

Solution

This can occur if the same Identity Store is not used by WebCenter Portal and the Discussions server; the user must exist in both stores if the store is not shared.

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 Events Prerequisites for Personal Events.
- Is the events server connection correct in the managed server? For more information, see 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.

Troubleshooting Issues with Users and Roles

For Oracle 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(UCMSpacesUtils.ja
va: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.

Troubleshooting Issues with Content Repositories

This section includes the following subsections:

- Documents Tools Unavailable in WebCenter Portal

Documents Tools 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 WebCenter Content Server, or the WebCenter Content Server does not contain some required WebCenter Portal data.

To diagnose the problem, follow these steps:

1. Check that the WebCenter 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.
    b. Select and edit the required connection.
    c. Ensure the **Active Connection** check box is selected.
    d. Ensure that **Content Administrator**, **Portal Server Identifier** and **Security Group** 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.
        - **Portal Server Identifier and Security Group** - both must be unique and not used by any other WebCenter Portal instance using the same WebCenter Content Server. If you change these values, ensure that both values are changed and not just one of them.
        - **Security Group** - must be 14 characters or less as it is used as a prefix for items created in WebCenter Content Server, which have a limit on the length of the item name.
    e. If you make changes, click **Test** to verify that the connection works.
    f. Click **OK** to save the connection.
    g. If you made changes, you must restart `WC_Portal`, 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 WebCenter Content Server or saving data on the WebCenter Content Server.

    For details, see Viewing and Configuring WebCenter Portal Logs.
4. If the log does not show any useful log information, increase the logging level for the WebCenter Content Server, and then restart WebCenter Portal to investigate the messages in more detail:
    a. i. 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, Viewing and Configuring WebCenter Portal Logs.
    ii. Restart WebCenter Portal.
    iii. View the logs again:

Troubleshooting Issues with Analytics

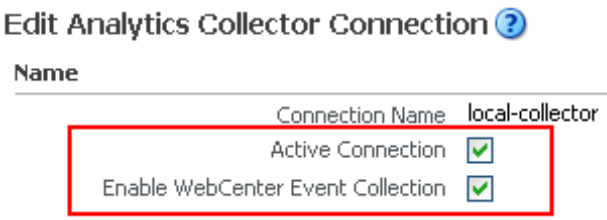
Problem

If users cannot see analytics in WebCenter Portal, verify the following:

Solution

- Check that the Analytics Collector configuration is correct and in particular that both **Enable WebCenter Event Collection** and **Active Connection** are both set. See Registering an Analytics Collector for Your Application.

Figure F-4 Enabling the Connection and Analytics Collection



If you make changes to the connection you must restart the managed server on which WebCenter Portal is deployed. For more information, see Starting and Stopping Managed Servers for WebCenter Portal Application Deployments.

- If WebCenter Portal 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.
- Perform the following:
  1. Shut down the managed server on which WebCenter Portal is deployed (`WC_Portal`).
  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.

Problem

Analytics Collector deployment fails when configured with the Pluggable Database (PDB), if the number of cursors in the database exceeds the maximum limit.

Solution

Try to increase the number of `open_cursors` in the Pluggable Database (PDB), for example 3000. The allowed value for the `open_cursors` is from 0 to 65535.

where, `open_cursors` specifies the maximum number of open cursors allowed per session.

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 Specifying the Notifications Channel Using WLST) or through the Fusion Middleware Control user interface (see 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 Managing Mail or Managing the SOA Connection for WebCenter Portal Membership Workflows.

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, see Subscribing to the Application, to Portals, and to Objects in *Using Portals in 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 Monitoring Oracle User Messaging Service in *Administering 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 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 Establishing and Managing Your Messaging Channels and Filters in *Using Portals in Oracle WebCenter Portal*.

Troubleshooting External Application Issues

This section contains common issues and workarounds related to external applications.

This section contains the following topic:

- Users Experience Password Lockout

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.

Troubleshooting Security Configuration Issues

This section includes the following subsections:

- WebCenter Portal Application Does Not Find Users in LDAP Provider
- Portal Created with Errors When Logged in as OID User
- Users Cannot Self-Register when WebCenter Portal Configured with Active Directory
- User Made Administrator Does Not Have Administrator Privileges
- Deploying the SAML SSO-specific Discussions EAR file Produces an Exception
- Configuring SAML Single Sign-on Produces 403 Error
- Impersonation Session Produces Error with OAM 11.1.2.2.0

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 Configuring the Identity Store. Restart the Administration Server and Managed Servers for the domain.

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_Portal` server.

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.

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`.

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.

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 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_Portal` 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, it would be Worklist Detail.)
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/configureWorklistDetail.py`
6. Test your configuration again. If all works well, you can disable SAML logging.

Impersonation Session Produces Error with OAM 11.1.2.2.0

Problem

An internal error is produced when switching between users in an impersonation session with OAM 11.1.2.2.0.

Solution

Check that:

- The `EnableImpersonation` flag in `oam-config.xml` is set to `true`
- The new OID identity store you created in the OAM 11.1.2.2 console does not contain spaces (for example, `Oracle Identity Store` rather than `OracleIdentityStore`)
- The UserPassword attribute for the new identity store is `userPassword`

Troubleshooting Issues with External Links

An external URL specified in a WebCenter Portal component, such as a link or wiki, may fail validation checks with the following message:

The URL entered is not available in the list of valid URLs. Contact your system administrator for the list of valid URLs.

For an external URL to be found valid, the system administrator must add it to the list of valid URLs in the `valid-link-url.xml` file. See Adding a List of Valid External URLs.

Troubleshooting Issues with Elasticsearch

This section includes the following topics.

- Profile Crawling Fails with 401 Error

Profile Crawling Fails with 401 Error

Problem

Profile crawling is a part of portal crawling and happens in batches. Sometimes one of the batches fails with 401 errors which can be observed in the diagnostic logs. You see the following kind of error logs:

```
Crawl failed for service
oracle.webcenter.peopleconnections.profile using crawl url
 http://host:port/rsscrawl?
context=&startIndex=77000&serviceId=oracle.webcenter.peopleconnections.profile&command=
GetData&jsonFormat with status 401
```

Solution

This issue appears when the service times out on the client-side. Increase the **Results Time Limit** value for the LDAP authentication provider in the WebLogic server console.

One of the reasons to see a 401 intermittently is due to LDAP client timeout issue. To determine if the reason is LDAP client timeout issue, do the following:

1. Log in to the WebLogic Server Administration Console.
2. Navigate to the managed server on which WebCenter Portal is deployed. Select **Environment** then, **Servers**, and then select the WebCenter Portal instance (`WC_Portal`).
3. Open the Debug tab, and expand **weblogic**, then **security**, and then **atn**.
4. Select the authentication logger that you want to enable and click **Enable**.
5. Schedule a crawl again. See Manually Starting a Full Crawl.

   This time you are expected to see a lot more debug logs in `WC_Portal.log`.

Around the time stamp you can see 401 in portal diagnostic log, check for server debug logs in `WC_Portal.log` and check if you see the following kind of error logs:

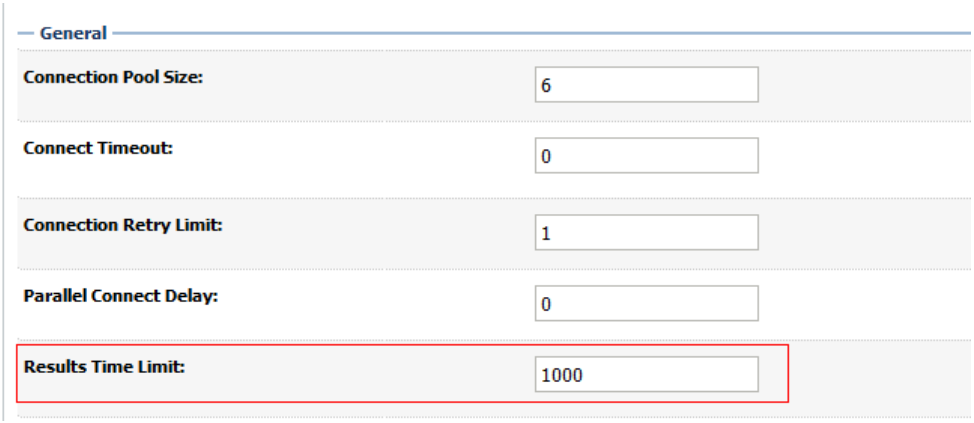
```
<weblogic.security.providers.authentication.LDAPAtnLoginModuleImpl.loginexception:java.
security.PrivilegedActionException:weblogic.security.providers.authentication.LoginServ
erUnavailableException:Time to complete operation exceeded
```

This confirms that it is a client-side timeout issue. To rectify this, increase the **Results Time Limit** value for the LDAP authentication provider in the WebLogic server console.

To update the Results Time Limit value:

1. Log in to the 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 LDAP authentication provider you are configuring.
5. Open Provider Specific tab and verify if the **Results Time Limit** is set to some value, other than 0.

Figure F-5 Specifying the Results Time Limit Value



6. Increase the **Results Time Limit** value and click Save.
7. Restart WebLogic Server for the changes to take effect.
8. schedule a crawl again. See Manually Starting a Full Crawl.