

Oracle® Fusion Middleware

Enterprise Deployment Guide for Oracle WebCenter

11g Release 1 (11.1.1)

E12037-02

October 2009

Oracle Fusion Middleware Enterprise Deployment Guide for Oracle WebCenter, 11g Release 1 (11.1.1)

E12037-02

Copyright © 2004, 2009, Oracle and/or its affiliates. All rights reserved.

Primary Author: Joe Paul

Contributing Author: Ron van de Crommert, John Bassett, Richard Delval, Joseph Garcia, Lypp-Tek Khoo-Ellis

Contributor: Janga Aliminati, Fermin Castro Alonso, Pradeep Bhat, Madhwa Chintarevula, Jordan Douglas, Paul Encarnacion, Jeni Ferns, Eileen He, Bharath Reddy, Michael Rhys, Praveen Sampath, Dhaval Shah, Sunita Sharma, Zhiming Zhu

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 software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software 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 which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software and documentation may provide access to or information on 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. 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.

Contents

Preface	xiii
Audience	xiii
Documentation Accessibility	xiii
Related Documents	xiv
Conventions	xiv
1 Enterprise Deployment Overview	
1.1 What is an Enterprise Deployment?	1-1
1.2 Terminology	1-2
1.3 Benefits of Oracle Recommendations	1-5
1.3.1 Built-in Security	1-5
1.3.2 High Availability	1-6
1.4 Hardware Requirements	1-6
1.5 Enterprise Deployment Reference Topology	1-6
1.5.1 Oracle Identity Management	1-8
1.5.2 Web Tier	1-8
1.5.2.1 Load Balancer Requirements	1-8
1.5.3 Application Tier	1-9
1.5.4 Data Tier	1-10
1.5.5 What to Install	1-10
1.5.6 Unicast Requirement	1-10
1.6 How to Use This Guide	1-11
1.6.1 Installation and Configuration Procedure	1-11
1.6.2 Overview of Installation Strategies	1-12
2 Database and Environment Preconfiguration	
2.1 Database	2-1
2.1.1 Setting Up the Database	2-1
2.1.1.1 Database Host Requirements	2-2
2.1.1.2 Supported Database Versions	2-2
2.1.1.3 Initialization Parameters	2-2
2.1.1.4 Database Services	2-3
2.1.2 Loading the Oracle Fusion Metadata Repository in the RAC Database	2-4
2.1.3 Backing Up the Database	2-5
2.2 Network	2-5

2.2.1	Virtual Server Names	2-5
2.2.1.1	wc.mycompany.com	2-5
2.2.1.2	admin.mycompany.com	2-6
2.2.1.3	soainternal.mycompany.com	2-6
2.2.1.4	wcinternal.mycompany.com	2-6
2.2.2	Load Balancers	2-6
2.2.2.1	Configuring the Load Balancer	2-6
2.2.3	IPs and Virtual IPs	2-8
2.2.4	Firewalls and Ports	2-9
2.3	Shared Storage and Recommended Directory Structure	2-11
2.3.1	Terminology for Directories and Directory Environment Variables	2-11
2.3.2	Recommended Locations for the Different Directories.....	2-12
2.3.3	Shared Storage Configuration.....	2-17
2.4	LDAP as Credential and Policy Store	2-18

3 Installing Oracle HTTP Server

3.1	Installing Oracle HTTP Server on WEBHOST1 and WEBHOST2	3-1
3.2	Validating Oracle HTTP Server Through the Load Balancer.....	3-3
3.3	Backing Up Oracle HTTP Server	3-4

4 Creating a Domain

4.1	Installing Oracle Fusion Middleware Home	4-2
4.1.1	Installing Oracle WebLogic Server.....	4-2
4.1.2	Installing Oracle Fusion Middleware for WebCenter	4-3
4.2	Backing Up the Installation	4-3
4.3	Enabling VIP1 in SOAHOST1	4-4
4.4	Running the Configuration Wizard on SOAHOST1 to Create a Domain	4-4
4.5	Creating a Separate Domain Directory for Managed Servers in the Same Node as the Administration Server	4-9
4.6	Creating boot.properties for the Administration Server on SOAHOST1	4-10
4.7	Starting the Administration Server on SOAHOST1	4-10
4.8	Validating the Administration Server.....	4-10
4.9	Applying the Java Required Files (JRF) Template to the WSM-PM_Cluster.....	4-11
4.10	Disabling Host Name Verification for the Oracle WebLogic Administration Server and the WLS_WSM1 Managed Server	4-11
4.11	Starting Node Manager on SOAHOST1	4-12
4.12	Starting and Validating the WLS_WSM1 Managed Server	4-12
4.13	Propagating the Domain Configuration to SOAHOST2 Using the unpack Utility	4-13
4.14	Disabling Host Name Verification for the WLS_WSM2 Managed Server	4-13
4.15	Starting Node Manager on SOAHOST2.....	4-14
4.16	Starting and Validating the WLS_WSM2 Managed Server	4-14
4.17	Configuring the Java Object Cache for Oracle WSM.....	4-14
4.18	Configuring Oracle HTTP Server for the Administration Server and the WLS_WSM-PM _n Managed Servers	4-16
4.19	Registering Oracle HTTP Server With WebLogic Server.....	4-17
4.20	Setting the Frontend URL for the Administration Console and Setting Redirection Preferences	4-17

4.21	Validating Access Through Oracle HTTP Server.....	4-18
4.22	Manually Failing Over the Administration Server to SOAHOST2.....	4-19
4.23	Validating Access to SOAHOST2 Through Oracle HTTP Server.....	4-20
4.24	Failing the Administration Server Back to SOAHOST1.....	4-20
4.25	Backing Up the Installation.....	4-21

5 Extending the Domain for SOA Components

5.1	Installing Oracle Fusion Middleware for SOA Home.....	5-2
5.2	Enabling SOAHOST1VHN2 and SOAHOST2VHN1.....	5-2
5.3	Extending the Domain for SOA Components.....	5-3
5.4	Restarting the Administration Server.....	5-8
5.5	Configuring Oracle Coherence for Deploying Composites.....	5-8
5.6	Running the soa-createUDD.py Script.....	5-10
5.7	Disabling Host Name Verification for the WLS_SOAn Managed Server.....	5-11
5.8	Restarting the Node Manager on SOAHOST1.....	5-11
5.9	Propagating the Domain Changes to the Managed Server Domain Directory.....	5-12
5.10	Starting the WLS_SOA1 Managed Server on SOAHOST1.....	5-12
5.11	Validating the WLS_SOA1 and WLS_WSM1 Managed Servers.....	5-12
5.12	Propagating the Domain Configuration to SOAHOST2 Using the unpack Utility.....	5-13
5.13	Extracting the XEngine Files in SOAHOST2.....	5-13
5.14	Restarting Node Manager on SOAHOST2.....	5-13
5.15	Starting and Validating the WLS_SOA2 Managed Server.....	5-13
5.16	Configuring Oracle HTTP Server for the WLS_SOAn Managed Servers.....	5-14
5.17	Validating Access Through Oracle HTTP Server.....	5-15
5.18	Setting the Frontend HTTP Host and Port.....	5-16
5.19	Configuring a Shared JMS Persistence Store.....	5-17
5.20	Configuring a Default Persistence Store for Transaction Recovery.....	5-17
5.21	Enabling High Availability for Oracle File and FTP Adapters.....	5-18
5.21.1	Using the Database Mutex Locking Operation.....	5-18
5.22	Scaling the Oracle Database Adapter.....	5-21
5.23	Backing Up the Installation.....	5-21

6 Extending the Domain for WebCenter Components

6.1	Installing Oracle Fusion Middleware Home.....	6-2
6.2	Extending the Domain for WebCenter Components.....	6-2
6.3	Restarting the Administration Server.....	6-6
6.4	Disabling Host Name Verification for the WebCenter Managed Servers.....	6-6
6.5	Starting Node Manager on SOAHOST1.....	6-6
6.6	Propagating the Domain Changes to the Managed Server Domain Directory.....	6-6
6.7	Propagating the Domain Configuration to SOAHOST2, WCHOST1, and WCHOST2 Using the unpack Utility.....	6-7
6.8	Starting the Node Manager on WCHOST1 and WCHOST2.....	6-7
6.9	Starting the WLS_Spaces1, WLS_Portlet1, and WLS_Services1 Managed Servers on WCHOST1.....	6-8
6.10	Validating the WLS_Spaces1, WLS_Portlet1, and WLS_Services1 Managed Servers.....	6-8
6.11	Starting the WLS_Spaces2, WLS_Portlet2, and WLS_Services2 Managed Servers on WCHOST2.....	6-9

6.12	Validating the WLS_Spaces2, WLS_Portlet2, and WLS_Services2 Managed Servers.....	6-9
6.13	Setting Up the Java Object Cache	6-9
6.14	Converting Discussions Forum from Multicast to Unicast	6-11
6.15	Configuring Clustering for Discussion Server	6-11
6.16	Configuring WS-Security for Discussion Server	6-12
6.16.1	Setting up the default Keystores.....	6-12
6.16.2	Distributing the Keystores.....	6-13
6.16.3	Set up Jive Server Properties	6-14
6.17	Configuring Wiki Server	6-14
6.18	Configuring Oracle HTTP Server for the WLS_Spaces <i>n</i> , WLS_Portlet <i>n</i> , and WLS_Services <i>n</i> Managed Servers on WCHOST2	6-15
6.19	Validating Access Through Oracle HTTP Server.....	6-17
6.20	Validating Access Through the Load Balancer	6-17
6.21	Backing Up the Installation	6-17

7 Setting Up Node Manager

7.1	About the Node Manager	7-1
7.2	Enabling Host Name Verification Certificates for Node Manager in SOAHOST1	7-1
7.2.1	Generating Self-Signed Certificates Using the utils.CertGen Utility	7-2
7.2.2	Creating an Identity Keystore Using the utils.ImportPrivateKey Utility.....	7-2
7.2.3	Creating a Trust Keystore Using the Keytool Utility	7-3
7.2.4	Configuring Node Manager to Use the Custom Keystores.....	7-4
7.3	Starting the Node Manager on SOAHOST1	7-4
7.4	Enabling Host Name Verification Certificates for the Node Manager in SOAHOST2	7-5
7.4.1	Generating Self-Signed Certificates Using the utils.CertGen Utility	7-5
7.4.2	Creating an Identity Keystore Using the "utils.ImportPrivateKey" Utility.....	7-6
7.4.3	Creating a Trust Keystore Using the keytool Utility	7-6
7.4.4	Configuring Node Manager to Use the Custom Keystores.....	7-7
7.5	Starting the Node Manager on SOAHOST2	7-8
7.6	Enabling Host Name Verification Certificates for Node Manager in WCHOST1 and WCHOST2	7-8

8 Configuring External Services

8.1	Configuring the Discussion Forum Connection.....	8-1
8.1.1	Configuring Discussions using WebCenter Spaces	8-1
8.1.2	Configuring Discussions via WLST	8-1
8.1.3	Creating a Discussions Server Connection for WebCenter From EM	8-2
8.2	Configuring the Instant Messaging and Presence (IMP) Server Connection.....	8-2
8.2.1	Configuring IMP using WebCenter Spaces	8-3
8.2.2	Configuring IMP via WLST.....	8-3
8.3	Configuring the Worklist and Workflow Server Connection	8-3
8.3.1	Configuring Worklist and Workflow via WebCenter Spaces	8-3
8.3.2	Configuring Worklist and Workflow via WLST	8-4
8.4	Configuring the Document Library Connection	8-4
8.4.1	Configuring the Document Library via WebCenter Spaces.....	8-4
8.4.2	Configuring the Document Library via WLST	8-5
8.5	Registering the Portlet Producers.....	8-5

8.5.1	Configuring the Portlet Producers via WebCenter Spaces.....	8-5
8.5.2	Configuring the Portlet Producers via WLST.....	8-5

9 Installing and Configuring Oracle Content Server

9.1	Prerequisites	9-1
9.2	Installing Oracle Content Server.....	9-1
9.3	Running the WebCenter Configuration Script	9-5
9.4	Configuring Oracle Content Server for Remote Oracle HTTP Server	9-6
9.4.1	Steps to Perform on the Content Server Machine.....	9-6
9.4.2	Steps to Perform on the Oracle HTTP Server Machine.....	9-6
9.5	Validating the Installation	9-8
9.6	Backing Up the Installation	9-8
9.7	Configuring Oracle Content Server for WebCenter	9-8
9.7.1	Configuring Oracle Content Server	9-8
9.7.1.1	Configuring the Identity Store.....	9-8
9.7.1.2	Enabling Full-Text Searching and Indexing	9-10
9.7.1.3	Configuring Secure Socket Layer (SSL) in Oracle Content Server	9-11
9.7.2	Configuring the Load Balancer.....	9-14
9.7.3	Installing Content Server Cluster	9-14
9.7.3.1	Installing the First Content Server Node	9-14
9.7.3.2	Installing the Second Content Server Node.....	9-15
9.7.3.3	Post-Installation Steps.....	9-15
9.7.3.4	Starting the Servers	9-16
9.8	Adding Another Oracle HTTP Server and Configuring a Load Balancer.....	9-16
9.8.1	Configuring the Second Oracle HTTP Server.....	9-16
9.8.2	Configuring the Load Balancer.....	9-16
9.8.3	Configuring the Content Server's HTTP Address	9-16
9.9	Reconfiguring the Content Server to Support RAC.....	9-17
9.10	Backing Up the Installation	9-17

10 Integration With Oracle Identity Management

10.1	Credential and Policy Store Configuration	10-1
10.1.1	Overview of Credential and Policy Store Configuration.....	10-1
10.1.2	Credential Store Configuration	10-2
10.1.2.1	Creating the LDAP Authenticator	10-2
10.1.2.2	Moving the WebLogic Administrator to LDAP.....	10-4
10.1.2.3	Reassociating the Domain Credential Store	10-7
10.1.3	Policy Store Configuration	10-7
10.1.3.1	Prerequisites to Using an LDAP-Based Policy Store.....	10-7
10.1.3.2	Cataloging Oracle Internet Directory Attributes	10-8
10.1.3.3	Reassociating the Domain Policy Store	10-8
10.1.4	Reassociation of Credentials and Policies	10-9
10.2	Oracle Access Manager Integration	10-10
10.2.1	Overview of Oracle Access Manager Integration	10-10
10.2.2	Prerequisites for Oracle Access Manager	10-10
10.2.3	Using the OAM Configuration Tool	10-10

10.2.3.1	Collecting the Information for the OAM Configuration Tool	10-11
10.2.3.2	Running the OAM Configuration Tool	10-11
10.2.3.3	Verifying Successful Creation of the Policy Domain and AccessGate	10-12
10.2.3.4	Updating the Host Identifier.....	10-13
10.2.3.5	Updating the WebGate Profile	10-13
10.2.3.6	Adding Additional Access Servers	10-14
10.2.3.7	Configure Delegated Form Authentication	10-15
10.2.4	Installing and Configuring WebGate.....	10-15
10.2.5	Setting Up the WebLogic Authenticators.....	10-20
10.2.5.1	Back Up Configuration Files	10-20
10.2.5.2	Setting Up the OAM ID Asserter	10-20
10.2.5.3	Setting the Order of Providers.....	10-20
10.2.6	Changing the Login Form for the Administration Console Application	10-21
10.3	Configuring WebCenter Applications	10-21
10.3.1	Configuring System Properties.....	10-21
10.3.2	Configuring the WebCenter Administrator Role.....	10-22
10.3.2.1	Granting the WebCenter Spaces Administrator Role Using WLST.....	10-22
10.3.2.2	Granting the WebCenter Spaces Administrator Role Using Fusion Middleware Control	10-23
10.3.3	Configuring Discussion Server and Wiki for OAM	10-23
10.3.3.1	Setting Up Discussions Server to Use OAM as SSO Provider	10-24
10.3.3.2	Adding a Wiki Page to WebCenter.....	10-24
10.4	Configuring WebCenter and BPEL Authentication.....	10-24
10.4.1	Set Authenticator	10-24
10.4.2	Set Role Members for BPMWorkflowAdmin Application Role in soa-infra	10-24
10.5	Backing Up the Installation	10-25

11 Managing the Topology

11.1	Monitoring the Topology.....	11-1
11.2	Configuring UMS Drivers	11-1
11.3	Scaling the Topology	11-2
11.3.1	Scaling Up the Topology (Adding Managed Servers to Existing Nodes).....	11-2
11.3.2	Scaling Out the Topology (Adding Managed Servers to New Nodes)	11-3
11.3.3	Scaling Out the Oracle Content Server (OCS) Topology	11-4
11.4	Performing Backups and Recoveries	11-5
11.5	Troubleshooting	11-7
11.5.1	Administration Server Fails to Start After a Manual Failover	11-7
11.5.2	Error While Activating Changes in Administration Console	11-7
11.5.3	OAM Configuration Tool Does Not Remove URLs	11-8
11.5.4	Portlet Unavailable After Database Failover	11-8
11.5.5	Redirecting of Users to Login Screen After Activating Changes in Administration Console	11-8
11.5.6	Redirecting of Users to Administration Console's Home Page After Activating Changes to OAM	11-9
11.5.7	Configured JOC Port Already in Use	11-9
11.5.8	Restoring a JMS Configuration	11-9
11.6	Best Practices	11-9

11.6.1	Preventing Timeouts for SQLNet Connections	11-10
11.6.2	Auditing	11-10

Index

Preface

This preface describes the audience, contents and conventions used in the *Oracle Fusion Middleware Enterprise Deployment Guide for WebCenter*.

Audience

This guide is intended for system administrators who are responsible for installing and configuring Oracle Fusion Middleware enterprise deployments.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at <http://www.oracle.com/accessibility/>.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Deaf/Hard of Hearing Access to Oracle Support Services

To reach Oracle Support Services, use a telecommunications relay service (TRS) to call Oracle Support at 1.800.223.1711. An Oracle Support Services engineer will handle technical issues and provide customer support according to the Oracle service request process. Information about TRS is available at <http://www.fcc.gov/cgb/consumerfacts/trs.html>, and a list of phone numbers is available at <http://www.fcc.gov/cgb/dro/trsphonebk.html>.

Related Documents

The following manuals in the Oracle Fusion Middleware documentation library provide additional information on the process of installing and configuring the Enterprise Deployment architectures:

- *Oracle Fusion Middleware Administrator's Guide*
- *Oracle Fusion Middleware Administrator's Guide for Oracle Virtual Directory*
- *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory*

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.
<i>italic</i>	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.

Enterprise Deployment Overview

This chapter provides an overview of the enterprise topology for Oracle WebCenter. It contains the following sections:

- [Section 1.1, "What is an Enterprise Deployment?"](#)
- [Section 1.2, "Terminology"](#)
- [Section 1.3, "Benefits of Oracle Recommendations"](#)
- [Section 1.4, "Hardware Requirements"](#)
- [Section 1.5, "Enterprise Deployment Reference Topology"](#)
- [Section 1.6, "How to Use This Guide"](#)

1.1 What is an Enterprise Deployment?

An enterprise deployment is an Oracle best practices blueprint based on proven Oracle high-availability and security technologies and recommendations for Oracle Fusion Middleware. The best practices described in these blueprints span all Oracle products across the entire technology stack: Oracle Database, Oracle Fusion Middleware, Oracle Applications, Oracle Collaboration Suite, and Fusion Middleware Control.

An Oracle Fusion Middleware enterprise deployment:

- considers various business service level agreements (SLA) to make high-availability best practices as widely applicable as possible
- leverages database grid servers and storage grid with low-cost storage to provide highly resilient, lower cost infrastructure
- uses results from extensive performance impact studies for different configurations to ensure that the high-availability architecture is optimally configured to perform and scale to business needs
- enables control over the length of time to recover from an outage and the amount of acceptable data loss from a natural disaster
- uses Oracle best practices and recommended architecture, which are independent of hardware and operating systems.

For more information on high availability practices, go to

<http://www.oracle.com/technology/Deploy/availability/htdocs/maa.htm>.

Note: This document focuses on enterprise deployments in Linux environments, but enterprise deployments can also be implemented in UNIX and Windows environments.

1.2 Terminology

This section identifies terms used to describe components in prior releases, and the terms to which they correlate in 11g Release 1 (11.1.1).

- **Oracle home:** An Oracle home contains installed files necessary to host a specific product. For example, the SOA Oracle home contains a directory that contains binary and library files for Oracle SOA Suite. An Oracle home resides within the directory structure of the Middleware home. Each Oracle home can be associated with multiple Oracle instances or Oracle WebLogic Server domains.
- **WebLogic Server home:** A WebLogic Server home contains installed files necessary to host a WebLogic Server. The WebLogic Server home directory is a peer of Oracle home directories and resides within the directory structure of the Middleware home.
- **Middleware home:** A Middleware home consists of the Oracle WebLogic Server home, and, optionally, one or more Oracle homes. A Middleware home can reside on a local file system or on a remote shared disk that is accessible through NFS.
- **Oracle instance:** An Oracle instance contains one or more active middleware system components, for example Oracle Web Cache, Oracle HTTP Server, or Oracle Internet Directory. You determine which components are part of an instance, either at install time or by creating and configuring an instance at a later time. An Oracle instance contains files that can be updated, such as configuration files, log files, temporary files.
- **failover:** When a member of a high availability system fails unexpectedly (unplanned downtime), in order to continue offering services to its consumers, the system undergoes a failover operation. If the system is an active-passive system, the passive member is activated during the failover operation and consumers are directed to it instead of the failed member. The failover process can be performed manually, or it can be automated by setting up hardware cluster services to detect failures and move cluster resources from the failed node to the standby node. If the system is an active-active system, the failover is performed by the load balancer entity serving requests to the active members. If an active member fails, the load balancer detects the failure and automatically redirects requests for the failed member to the surviving active members. See *Oracle Fusion Middleware High Availability Guide* for information on active-active and active-passive systems.
- **failback:** After a system undergoes a successful failover operation, the original failed member can be repaired over time and be re-introduced into the system as a standby member. If desired, a failback process can be initiated to activate this member and deactivate the other. This process reverts the system back to its pre-failure configuration.
- **hardware cluster:** A hardware cluster is a collection of computers that provides a single view of network services (for example: an IP address) or application services (for example: databases, Web servers) to clients of these services. Each node in a hardware cluster is a standalone server that runs its own processes. These processes can communicate with one another to form what looks like a single system that cooperatively provides applications, system resources, and data to users.

A hardware cluster achieves high availability and scalability through the use of specialized hardware (cluster interconnect, shared storage) and software (health monitors, resource monitors). (The cluster interconnect is a private link used by the hardware cluster for heartbeat information to detect node death.) Due to the need for specialized hardware and software, hardware clusters are commonly provided by hardware vendors such as Sun, HP, IBM, and Dell. While the number of nodes that can be configured in a hardware cluster is vendor dependent, for the purpose of Oracle Fusion Middleware high availability, only two nodes are required. Hence, this document assumes a two-node hardware cluster for high availability solutions employing a hardware cluster.

- **cluster agent:** The software that runs on a node member of a hardware cluster that coordinates availability and performance operations with other nodes. Clusterware provides resource grouping, monitoring, and the ability to move services. A cluster agent can automate the service failover.
- **clusterware:** A software that manages the operations of the members of a cluster as a system. It allows one to define a set of resources and services to monitor via a heartbeat mechanism between cluster members and to move these resources and services to a different member in the cluster as efficiently and transparently as possible.
- **shared storage:** Shared storage is the storage subsystem that is accessible by all the machines in the EDG domain. Among other things, the following is located on the shared disk:
 - Middleware Home software
 - AdminServer Domain Home
 - JMS
 - Tlogs (where applicable)

Managed Server homes can also be optionally located in the shared disk. The shared storage can be a Network Attached Storage (NAS), a Storage Area Network (SAN) or any other storage system that multiple nodes can access simultaneously and can read-write.

- **primary node:** The node that is actively running an Oracle Fusion Middleware instance at any given time and has been configured to have a backup/secondary node. If the primary node fails, Oracle Fusion Middleware instance is failed over to the secondary node. This failover can be manual or automated using the Clusterware for Administration Server. For a server migration based scenario, WebLogic Whole Server Migration is used for automated failover.
- **secondary node:** The node that is the backup node for an Oracle Fusion Middleware instance. This is where the active instance fails over when the primary node is no longer available. See the definition for primary node in this section.
- **network host name:** Network host name is a name assigned to an IP address either through the `/etc/hosts` file or through DNS resolution. This name is visible in the network that the machine to which it refers to is connected. Often, the network host name and physical host name are identical. However, each machine has only one physical host name but may have multiple network host names. Thus, a machine's network host name may not always be its physical host name.
- **physical host name:** This guide differentiates between the terms physical host name and network host name. This guide uses physical host name to refer to the

"internal name" of the current machine. On UNIX, this is the name returned by the `hostname` command.

Physical host name is used by Oracle Fusion Middleware to reference the local host. During installation, the installer automatically retrieves the physical host name from the current machine and stores it in the Oracle Fusion Middleware configuration metadata on disk.

- **physical IP:** Physical IP refers to the IP of a machine on the network. In almost all cases, it is normally associated with the physical host name of the machine (see the definition of the physical host name). In contrast to a virtual IP, it is always associated with the same machine when on a network.
- **switchover:** During normal operation, active members of a system may require maintenance or upgrading. A switchover process can be initiated to allow a substitute member to take over the workload performed by the member that requires maintenance or upgrading, which undergoes planned downtime. The switchover operation ensures continued service to consumers of the system.
- **switchback:** When a switchover operation is performed, a member of the system is deactivated for maintenance or upgrading. When the maintenance or upgrading is completed, the system can undergo a switchback operation to activate the upgraded member and bring the system back to the pre-switchover configuration.
- **virtual host name:** Virtual host name is a network addressable host name that maps to one or more physical machines via a load balancer or a hardware cluster. For load balancers, the name "virtual server name" is used interchangeably with virtual host name in this book. A load balancer can hold a virtual host name on behalf of a set of servers, and clients communicate indirectly with the machines using the virtual host name. A virtual host name in a hardware cluster is a network host name assigned to a cluster virtual IP. Because the cluster virtual IP is not permanently attached to any particular node of a cluster, the virtual host name is not permanently attached to any particular node either.

Note: Whenever the term "virtual host name" is used in this document, it is assumed to be associated with a virtual IP address. In cases where just the IP address is needed or used, it will be explicitly stated.

- **virtual IP:** Also, cluster virtual IP and load balancer virtual IP. Generally, a virtual IP can be assigned to a hardware cluster or load balancer. To present a single system view of a cluster to network clients, a virtual IP serves as an entry point IP address to the group of servers which are members of the cluster. A virtual IP can be assigned to a server load balancer or a hardware cluster.

A hardware cluster uses a cluster virtual IP to present to the outside world the entry point into the cluster (it can also be set up on a standalone machine). The hardware cluster's software manages the movement of this IP address between the two physical nodes of the cluster while clients connect to this IP address without the need to know which physical node this IP address is currently active on. In a typical two-node hardware cluster configuration, each machine has its own physical IP address and physical host name, while there could be several cluster IP addresses. These cluster IP addresses float or migrate between the two nodes. The node with current ownership of a cluster IP address is active for that address.

A load balancer also uses a virtual IP as the entry point to a set of servers. These servers tend to be active at the same time. This virtual IP address is not assigned to

any individual server but to the load balancer which acts as a proxy between servers and their clients.

1.3 Benefits of Oracle Recommendations

The Oracle Fusion Middleware configurations discussed in this guide are designed to ensure security of all invocations, maximize hardware resources, and provide a reliable, standards-compliant system for enterprise computing with a variety of applications.

- [Section 1.3.1, "Built-in Security"](#)
- [Section 1.3.2, "High Availability"](#)

The security and high availability benefits of the Oracle Fusion Middleware configurations are realized through isolation in firewall zones and replication of software components.

1.3.1 Built-in Security

The Enterprise Deployment architectures are secure because every functional group of software components is isolated in its own DMZ, and all traffic is restricted by protocol and port. The following characteristics ensure security at all needed levels, as well as a high level of standards compliance:

- Configure external load balancers to redirect all external communication received on port 80 to port 443.

Note: The Oracle Technology Network (<http://www.oracle.com/technology/index.html>) provides a list of validated load balancers and their configuration at http://www.oracle.com/technology/products/ias/hi_av/Tested_LBR_FW_SSLSslAccel.html.

- Communication from external clients does not go beyond the Load Balancing Router level.
- No direct communication from the Load Balancing Router to the data tier is allowed.
- Components are separated in different protection zones: the web tier, application tier, and the data tier.
- Direct communication between two firewalls at any one time is prohibited.
- If a communication begins in one firewall zone, it must end in the next firewall zone.
- Oracle Internet Directory is isolated in the data tier.
- Identity Management components are in a separate subnet.
- All communication between components across protection zones is restricted by port and protocol, according to firewall rules.

1.3.2 High Availability

The enterprise deployment architectures are highly available, because each component or functional group of software components is replicated on a different computer, and configured for component-level high availability.

1.4 Hardware Requirements

Typical hardware requirements for the Enterprise Deployment on Linux operating systems are listed in [Table 1-1](#). The memory figures represent the memory required to install and run an Oracle Fusion Middleware server; however, for most production sites, you should configure at least 4 GB of physical memory.

For detailed requirements, or for requirements for other platforms, see the Oracle Fusion Middleware Installation Guide for that platform.

Table 1-1 Typical Hardware Requirements

Server	Processor	Disk	Memory	TMP Directory	Swap
Database	4 or more X Pentium, 1.5 GHz or greater	nXm n = number of disks, at least 4 (striped as one disk) m = size of the disk (minimum of 30 GB)	6-8 GB	Default	Default
WEBHOST n	2 or more X Pentium, 1.5 GHz or greater	10 GB	4 GB	Default	Default
SOAHOST n	2 or more X Pentium, 1.5 GHz or greater	10 GB ¹	4 GB	Default	Default
WCHOST n	2 or more X Pentium, 1.5 GHz or greater	10 GB	4 GB	Default	Default

¹ For a shared storage MW_HOME configuration, two installations suffice by making a total of 20 GB independently of the number of slots.

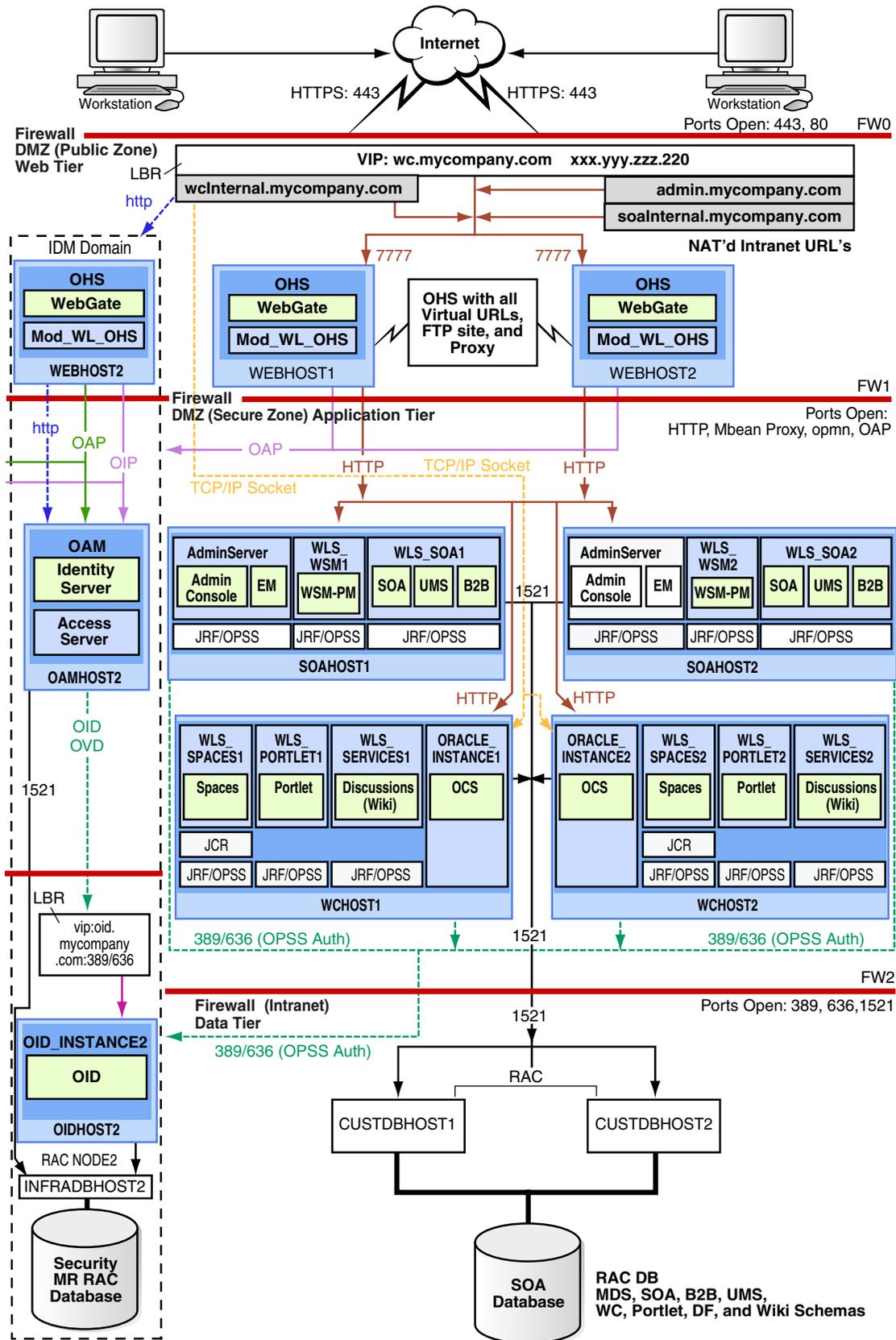
Note: You must perform the appropriate capacity planning to determine the number of nodes, CPU, and memory requirements for each node depending on the specific system's load as well as the throughput and response requirements. These will vary for each application or custom SOA system being used.

1.5 Enterprise Deployment Reference Topology

The instructions and diagrams in this guide describe a reference topology, to which variations may be applied.

This guide provides configuration instructions for a reference enterprise topology that uses Oracle WebCenter with Oracle Access Manager, as shown in [Figure 1-1](#).

Figure 1-1 MyWCCompany Topology with Oracle Access Manager



This section covers these topics:

- [Section 1.5.1, "Oracle Identity Management"](#)
- [Section 1.5.2, "Web Tier"](#)
- [Section 1.5.3, "Application Tier"](#)
- [Section 1.5.4, "Data Tier"](#)
- [Section 1.5.5, "What to Install"](#)
- [Section 1.5.6, "Unicast Requirement"](#)

1.5.1 Oracle Identity Management

Integration with the Oracle Identity Management system is an important aspect of the enterprise deployment architecture. This integration provides features such as single sign-on, integration with OPSS, centralized identity and credential store, authentication for the WebLogic domain, and so on. The IDM EDG is separate from this EDG and exists in a separate domain by itself. For more information on identity management in an enterprise deployment context, see *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*.

The primary interface to the IDM EDG is the LDAP traffic to the LDAP servers, the OAP (Oracle Access Protocol) to the OAM Access Servers, and the HTTP redirection of authentication requests.

1.5.2 Web Tier

Nodes in the web tier are located in the DMZ public zone.

In this tier, two nodes WEBHOST1 and WEBHOST2 run Oracle HTTP Server configured with WebGate and mod_wl_ohs.

Through mod_wl_ohs, which allows requests to be proxied from Oracle HTTP Server to WebLogic Server, Oracle HTTP Server forwards the requests to WebLogic Server running in the application tier.

WebGate (which is an Oracle Access Manager component) in Oracle HTTP Server uses Oracle Access Protocol (OAP) to communicate with Oracle Access Manager running on OAMHOST2, in the Identity Management DMZ. WebGate and Oracle Access Manager are used to perform operations such as user authentication.

The web tier also includes a load balancer router to handle external requests. External requests are sent to the virtual host names configured on the load balancer. The load balancer then forwards the requests to Oracle HTTP Server.

The WebGate module in Oracle HTTP Server uses Oracle Access Protocol (OAP) to communicate with Oracle Access Manager to perform operations such as querying user groups.

On the firewall protecting the web tier, only the HTTP ports are open: 443 for HTTPS and 80 for HTTP.

1.5.2.1 Load Balancer Requirements

This enterprise topology uses an external load balancer. This external load balancer should have the following features:

- Ability to load-balance traffic to a pool of real servers through a virtual host name: Clients access services using the virtual host name (instead of using actual host

names). The load balancer can then load balance requests to the servers in the pool.

- Port translation configuration should be possible so that incoming requests on the virtual host name and port are directed to a different port on the backend servers.
- Monitoring of ports on the servers in the pool to determine availability of a service.
- Virtual servers and port configuration: Ability to configure virtual server names and ports on your external load balancer, and the virtual server names and ports must meet the following requirements:
 - The load balancer should allow configuration of multiple virtual servers. For each virtual server, the load balancer should allow configuration of traffic management on more than one port. For example, for Oracle HTTP Server in the web tier, the load balancer needs to be configured with a virtual server and ports for HTTP and HTTPS traffic.
 - The virtual server names must be associated with IP addresses and be part of your DNS. Clients must be able to access the external load balancer through the virtual server names.
- Ability to detect node failures and immediately stop routing traffic to the failed node.
- Fault-tolerant mode: It is highly recommended that you configure the load balancer to be in fault-tolerant mode.
- It is highly recommended that you configure the load balancer virtual server to return immediately to the calling client when the backend services to which it forwards traffic are unavailable. This is preferred over the client disconnecting on its own after a timeout based on the TCP/IP settings on the client machine.
- Sticky routing capability: Ability to maintain sticky connections to components. Examples of this include cookie-based persistence, IP-based persistence, and so on.
- The load balancer should be able to terminate SSL requests at the load balancer and forward traffic to the backend real servers using the equivalent non-SSL protocol (for example, HTTPS to HTTP). Typically, this feature is called SSL acceleration and it is required for this EDG.

1.5.3 Application Tier

Nodes in the application tier are located in the DMZ secure zone.

SOAHOST1 and SOAHOST2 run the Oracle WebLogic Server Administration Console and Oracle Enterprise Manager Fusion Middleware Control, but in an active-passive configuration. You can fail over the Administration Server manually (see [Section 4.22, "Manually Failing Over the Administration Server to SOAHOST2"](#)); alternatively you can configure the Oracle WebLogic Server Administration Console with CFC/CRS to fail over automatically on a separate hardware cluster (not shown in this architecture).

Oracle WebCenter components such as WebCenter Spaces, Portlets, Wiki/Discussion Forum, and WebCenter custom applications run on WCHOST1 and WCHOST2 in an active-active configuration. Typically the managed servers are called WLS_Spaces, WLS_Portlet, and WLS_Services (for Wiki/Discussion Forum). You can also create a managed server to run WebCenter custom applications, depending on the requirements of the custom applications.

WCHOST1 and WCHOST2 also run Oracle Content Server (OCS). OCS is configured in an active-active manner.

If you are also running SOA components in this topology, SOAHOST1 and SOAHOST2 run WebLogic Server configured with the WLS_SOA and WLS_WSM managed servers, which run SOA components. These components are configured in an active-active manner.

Oracle Web Services Manager (Oracle WSM) provides a policy framework to manage and secure Web services in the EDG topology. WSM Policy Manager also runs in active-active configuration in two additional WebLogic Servers.

On the firewall protecting the application tier, the HTTP ports, OAP port, and proxy port are open. The OAP port is for the WebGate module running in Oracle HTTP Server in the web tier to communicate with Oracle Access Manager. Applications requiring external HTTP access use Oracle HTTP Server as the proxy. (The proxy on the Oracle HTTP Server must be enabled to allow this access.)

1.5.4 Data Tier

Nodes in the data tier are located in the most secured network zone (the intranet).

In this tier, an RAC database runs on the nodes CUSTDBHOST1 and CUSTDBHOST2. The database contains the schemas needed by the SOA Oracle WebCenter components. The WebCenter and SOA components running in the application tier access this database.

On the firewall protecting the data tier, the database listener port (typically, 1521) is required to be open. The LDAP ports (typically, 389 and 636) are also required to be open for the traffic accessing the LDAP storage in the IDM EDG.

1.5.5 What to Install

Table 1–2 identifies the source for installation of each software component. For more information, see *Oracle Fusion Middleware Installation Guide for Oracle SOA Suite* and *Oracle Fusion Middleware Installation Guide for Oracle WebCenter*.

Table 1–2 Components and Installation Sources

Component	Distribution Medium
Oracle Database 10g or 11g	Oracle Database CD (in 10g series, 10.2.0.4 or higher; in 11g series, 11.1.0.7 or higher)
Repository Creation Utility (RCU)	Oracle Fusion Middleware Repository Creation Utility 11g (11.1.1.1.0) DVD
Oracle WebLogic Server (WLS)	Oracle Weblogic Server (10.3.1) DVD
Oracle HTTP Server	Oracle Fusion Middleware WebTier and Utilities 11g (11.1.1.1.0) DVD
Oracle SOA Suite	Oracle SOA Suite 11g (11.1.1.1.0) DVD
Oracle Access Manager 10g Webgate	Oracle Access Manager 10g Webgates (10.1.4.3.0) DVD ; OAM OHS 11g webgates per platform
Oracle Virtual Directory (OVD)	Oracle Identity Management 11g (11.1.1.1.0) DVD

1.5.6 Unicast Requirement

Oracle recommends that the nodes in the myWCCompany topology communicate using unicast. Unlike multicast communication, unicast does not require cross-network configuration and it reduces potential network errors that can occur from multicast address conflicts as well.

The following considerations apply when using unicast to handle cluster communications:

- All members of a WebLogic cluster must use the same message type. Mixing between multicast and unicast messaging is not allowed.
- Individual cluster members cannot override the cluster messaging type.
- The entire cluster must be shut down and restarted to change the message modes (from unicast to multicast or from multicast to unicast).
- JMS topics configured for multicasting can access WebLogic clusters configured for unicast because a JMS topic publishes messages on its own multicast address that is independent of the cluster address. However, the following considerations apply:
 - The router hardware configurations that allow unicast clusters may not allow JMS multicast subscribers to work.
 - JMS multicast subscribers need to be in a network hardware configuration that allows multicast accessibility. (That is, JMS subscribers must be in a multicast-enabled network to access multicast topics.)

1.6 How to Use This Guide

This section covers the following topics:

- [Section 1.6.1, "Installation and Configuration Procedure"](#)
- [Section 1.6.2, "Overview of Installation Strategies"](#)

1.6.1 Installation and Configuration Procedure

[Table 1–3](#) summarizes the process by which you install and configure WebCenter. Follow the procedures indicated in the first column, in the order shown, for the chosen configuration.

Note: This document focuses on enterprise deployments in Linux environments, but enterprise deployments can also be implemented in UNIX and Windows environments.

Table 1–3 *WebCenter Installation Procedures*

Perform the steps in...	To configure a domain with only Admin Server and WSM-PM	To configure a domain with Admin Server, WSM-PM, and to extend a domain with a SOA cluster	To configure a domain with Admin Server, WCM-PM, SOA cluster, and WebCenter
Chapter 2, "Database and Environment Preconfiguration"	Yes	Yes	Yes
Chapter 3, "Installing Oracle HTTP Server"	Yes	Yes	Yes
Chapter 4, "Creating a Domain"	Yes	Yes	Yes
Chapter 5, "Extending the Domain for SOA Components"	No	Yes	Yes
Chapter 6, "Extending the Domain for WebCenter Components"	No	No	Yes

Table 1–3 (Cont.) WebCenter Installation Procedures

Perform the steps in...	To configure a domain with only Admin Server and WSM-PM	To configure a domain with Admin Server, WSM-PM, and to extend a domain with a SOA cluster	To configure a domain with Admin Server, WCM-PM, SOA cluster, and WebCenter
Chapter 7, "Setting Up Node Manager"	Recommended (optional depending on the type of security required for the application tier)	Recommended (optional depending on the type of security required for the application tier)	Recommended (optional depending on the type of security required for the application tier)
Chapter 8, "Configuring External Services"	No	No	Yes
Chapter 9, "Installing and Configuring Oracle Content Server"	No	No	Yes

1.6.2 Overview of Installation Strategies

The Configuration Wizard enables you to extend the Oracle WebLogic domain by adding only the needed components; rather than using the Configuration Wizard to create SOA components and the Oracle WebCenter components along with the domain that includes the Administration Server, Enterprise Manager, and WSM-PM in a single pass, you can instead create the domain and its Administration Server, Enterprise Manager, and WSM-PM in one pass of the Configuration Wizard and then extend the domain by adding only the SOA components (or if needed, only the WebCenter components) in a subsequent pass. Using this incremental approach, you can verify the installation of the servers and perform specific validations after each pass of the Configuration Wizard. In general, Oracle recommends the following approach:

1. Run a first pass of the Configuration Wizard to install the Administration Server, Enterprise Manager, and WSM-PM (described in [Chapter 4, "Creating a Domain"](#)).
2. Optionally, run a second pass of the Configuration Wizard to install the SOA components (described in [Chapter 5, "Extending the Domain for SOA Components"](#)).
3. Run a third pass to install the WebCenter components (described in [Chapter 6, "Extending the Domain for WebCenter Components"](#)).

Oracle recommends this modular approach in order to facilitate the verification of individual components one by one. This building block approach simplifies the troubleshooting during the setup process and facilitates the configuration in smaller steps.

Some variation from the above topology is possible. For example, if a deployment chooses to install WebCenter alone, then only sections applicable extend with WebCenter need to be followed. Also, in this case, it is expected that the Admin Server will exist on WCHOST1 instead and the instructions on creating the domain should be modified appropriately.

Database and Environment Preconfiguration

This chapter describes database and network environment preconfiguration required by the WebCenter enterprise topology. This chapter contains the following sections:

- [Section 2.1, "Database"](#)
- [Section 2.2, "Network"](#)
- [Section 2.3, "Shared Storage and Recommended Directory Structure"](#)

2.1 Database

For the WebCenter enterprise topology, the database contains the Oracle Fusion Middleware Repository, which is a collection of schemas used by various Oracle Fusion Middleware components, such as the WebCenter components, and OWSM. This database is separate from the Identity Management database, which is used in Identity Management EDG by components such as Oracle Internet Directory, DIP, and so on.

You must install the Oracle Fusion Middleware Repository before you can configure the Oracle Fusion Middleware components. You install the Oracle Fusion Middleware metadata repository into an existing database using the Repository Creation Utility (RCU), which is available from the RCU DVD or from the location listed in [Table 1-2](#). For the enterprise topology, a Real Application Clusters (RAC) database is highly recommended.

When you configure the WebCenter components, the configuration wizard will prompt you to enter the information for connecting to the database that contains the metadata repository.

This section covers the following topics:

- [Section 2.1.1, "Setting Up the Database"](#)
- [Section 2.1.2, "Loading the Oracle Fusion Metadata Repository in the RAC Database"](#)
- [Section 2.1.3, "Backing Up the Database"](#)

2.1.1 Setting Up the Database

Before loading the metadata repository into your database, check that the database meets the requirements described in these subsections:

- [Section 2.1.1.1, "Database Host Requirements"](#)
- [Section 2.1.1.2, "Supported Database Versions"](#)

- [Section 2.1.1.3, "Initialization Parameters"](#)
- [Section 2.1.1.4, "Database Services"](#)

2.1.1.1 Database Host Requirements

On the hosts CUSTDBHOST1 and CUSTDBHOST2 in the data tier, note the following requirements:

- **Oracle Clusterware**
For 11g Release 1 (11.1) for Linux, refer to the *Oracle Clusterware Installation Guide for Linux*.
- **Oracle Real Application Clusters**
For 11g Release 1 (11.1) for Linux, refer to the *Oracle Real Application Clusters Installation Guide for Linux*. For 10g Release 2 (10.2) for Linux, refer to *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Installation Guide*.
- **Automatic Storage Management (optional)**
ASM gets installed for the node as a whole. It is recommended that you install it in a separate Oracle Home from the Database Oracle Home. This option comes in at runInstaller. In the Select Configuration page, select the Configure Automatic Storage Management option to create a separate ASM home.

2.1.1.2 Supported Database Versions

The following versions of the Oracle Database are supported:

- **Oracle Database 11g (11.1.0.7 or higher)**. It is recommended to use the Enterprise Edition version, particularly if a Disaster Recovery site is planned or expected.
- **Oracle Database 10g (10.2.0.4 or higher)**. Oracle Database 10g Express Edition (Oracle Database XE) is not supported in Oracle Fusion Middleware 11g release 1. It is recommended to use the Enterprise Edition version, particularly if a Disaster Recovery site is planned or expected.

To check the release of your database, you can query the PRODUCT_COMPONENT_VERSION view as follows:

```
SQL> SELECT VERSION FROM SYS.PRODUCT_COMPONENT_VERSION WHERE
PRODUCT LIKE 'Oracle%';
```

2.1.1.3 Initialization Parameters

Ensure that the following initialization parameter is set to the required minimum value. It is checked by Repository Creation Assistant.

Table 2-1 Required Initialization Parameters

Configuration	Parameter	Required Value	Parameter Class
SOA	PROCESSES	300 or greater	Static
WC	PROCESSES	300 or greater	Static
SOA and WC	PROCESSES	600 or greater	Static

To check the value of the initialization parameter using SQL*Plus, you can use the SHOW PARAMETER command.

As the SYS user, issue the SHOW PARAMETER command as follows:

```
SQL> SHOW PARAMETER processes
```

Set the initialization parameter using the following command:

```
SQL> ALTER SYSTEM SET processes=300 SCOPE=SPFILE;
```

Restart the database.

Note: The method that you use to change a parameter's value depends on whether the parameter is static or dynamic, and on whether your database uses a parameter file or a server parameter file. See the *Oracle Database Administrator's Guide* for details on parameter files, server parameter files, and how to change parameter values.

2.1.1.4 Database Services

Oracle recommends using the Oracle Enterprise Manager Cluster Managed Services Page to create database services that client applications will use to connect to the database. For complete instructions on creating database services, see the chapter on workload management in the *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide*.

You can also use SQL*Plus to configure this using the following instructions:

1. Use the CREATE_SERVICE subprogram to create the `wcedg.mycompany.com` database service. Log on to SQL*Plus as the sysdba user and run the following command:

```
SQL> EXECUTE DBMS_SERVICE.CREATE_SERVICE
(SERVICE_NAME => 'wcedg.mycompany.com',
NETWORK_NAME => 'wcedg.mycompany.com',
);
```

2. Add the service to the database and assign it to the instances using `srvctl`:

```
prompt> srvctl add service -d wcdb -s wcedg -r wcdb1,wcdb2
```

3. Start the service using `srvctl`:

```
prompt> srvctl start service -d wcdb -s wcedg
```

Note: For more information about the SRVCTL command, see the *Oracle Real Application Clusters Administration and Deployment Guide*.

Oracle recommends that a specific database service be used for a product suite even when they share the same database. It is also recommended that the database service used is different than the default database service. In this case, the database is `wcdb.mycompany.com` and the default service is one with the same name. The WebCenter install is configured to use the service `wcedg.mycompany.com`. It is recommended that a service named `soaedg.mycompany.com` is used for SOA.

2.1.2 Loading the Oracle Fusion Metadata Repository in the RAC Database

To load the Oracle Fusion Middleware Repository into a database, complete these steps:

1. Start Repository Creation Utility (RCU), which is available from the RCU DVD or from the location listed in [Table 1-2](#), by first inserting the RCU DVD.
2. Start RCU from the *bin* directory:

```
./rcu
```
3. In the Welcome screen, click **Next**.
4. In the Create Repository screen, select **Create** to load component schemas into a database. Click **Next**.
5. In the Database Connection Details screen, enter connect information for your database:
 - **Database Type:** select **Oracle Database**.
 - **Host Name:** Enter the name of the node that is running the database. For the RAC database, specify the VIP name or one of the node names as the host name: CUSTDBHOST1-VIP.
 - **Port:** Enter the port number for the database: 1521.
 - **Service Name:** Enter the service name of the database:wcedg.mycompany.com
 - **Username:** SYS
 - **Password:** Enter the password for the SYS user.
 - **Role:** SYSDBAClick **Next**.
6. If you get this warning message: The database you are connecting is with non-UTF8 charset, if you are going to use this database for multilingual support, you may have data loss. If you are not using for multilingual support you can continue, otherwise we strongly recommend using UTF-8 database.
Click **Ignore** or **Stop**.
7. In the Select Components screen, do the following:
 - Select **Create a New Prefix**, and enter a prefix to use for the database schemas. Example: DEV or PROD. Prefixes are used to create logical groupings of multiple repositories in a database. For more information, see *Oracle Fusion Middleware Repository Creation Utility User's Guide*.

Tip: Note the name of the schema because the upcoming steps require this information.
 - Select the following:
 - WebCenter Infrastructure:
 - **WebCenter Suite**

Note: Note that this will auto-select Metadata Services as well.

Click **Next**.

8. In the Schema Passwords screen, enter passwords for the main and additional (auxiliary) schema users, and click **Next**.
9. In the Map Tablespaces screen, choose the tablespaces for the selected components, and click **Next**.
10. In the Summary screen, click **Create**.
11. In the Completion Summary screen, click **Close**.

2.1.3 Backing Up the Database

After you have loaded the metadata repository in your database, you should make a backup.

Backing up the database is for the explicit purpose of quick recovery from any issue that may occur in the further steps. You can choose to use your backup strategy for the database for this purpose or simply take a backup using OS tools or RMAN for this purpose. It is recommended to use Oracle Recovery Manager for the database, particularly if the database was created using Oracle ASM. If possible, a cold backup using operating system tools such as tar can also be performed.

2.2 Network

This section covers the following topics:

- [Section 2.2.1, "Virtual Server Names"](#)
- [Section 2.2.2, "Load Balancers"](#)
- [Section 2.2.3, "IPs and Virtual IPs"](#)
- [Section 2.2.4, "Firewalls and Ports"](#)

2.2.1 Virtual Server Names

The WebCenter enterprise topology uses the following virtual server names:

- [Section 2.2.1.1, "wc.mycompany.com"](#)
- [Section 2.2.1.2, "admin.mycompany.com"](#)
- [Section 2.2.1.3, "soainternal.mycompany.com"](#)
- [Section 2.2.1.4, "wcinternal.mycompany.com"](#)

Ensure that the virtual server names are associated with IP addresses and are part of your DNS. The nodes running Oracle Fusion Middleware must be able to resolve these virtual server names.

2.2.1.1 wc.mycompany.com

`wc.mycompany.com` is a virtual server name that acts as the access point for all HTTP traffic to the runtime SOA and WebCenter components, such as soa-infra, workflow, and B2B. Traffic to SSL is configured. Clients access this service using the address `wc.mycompany.com:443`. This virtual server is defined on the load balancer.

2.2.1.2 admin.mycompany.com

admin.mycompany.com is a virtual server name that acts as the access point for all internal HTTP traffic that is directed to administration services such as WebLogic Administration Server Console and Oracle Enterprise Manager.

The incoming traffic from clients is not SSL-enabled. Clients access this service using the address admin.mycompany.com:80 and the requests are forwarded to port 7777 on WEBHOST1 and WEBHOST2.

This virtual server is defined on the load balancer.

2.2.1.3 soainternal.mycompany.com

soainternal.mycompany.com is a virtual server name used for internal invocations of SOA services. This url is not exposed to the internet and is only accessible from the intranet. (For SOA systems, users can set this while modeling composites or at runtime with the appropriate EM/MBeans, as the url to be used for internal services invocations.)

The incoming traffic from clients is not SSL-enabled. Clients access this service using the address soainternal.mycompany.com:80 and the requests are forwarded to port 7777 on WEBHOST1 and WEBHOST2.

This virtual server is defined on the load balancer.

2.2.1.4 wcinternal.mycompany.com

wcinternal.mycompany.com is a virtual server name used for internal invocations like callbacks and internal access to services. This url is not exposed to the internet and is only accessible from the intranet.

The incoming traffic from clients is not SSL-enabled. Clients access this service using the address wcinternal.mycompany.com:80 and the requests are forwarded to port 7777 on WEBHOST1 and WEBHOST2.

This virtual server is defined on the load balancer.

2.2.2 Load Balancers

This enterprise topology uses an external load balancer. For more information on load balancers, see [Section 1.5.2, "Web Tier."](#)

Note: The Oracle Technology Network (<http://www.oracle.com/technology/index.html>) provides a list of validated load balancers and their configuration at http://www.oracle.com/technology/products/ias/hi_av/Tested_LBR_FW_SSLAccel.html.

2.2.2.1 Configuring the Load Balancer

To configure the load balancer, complete these steps:

1. Create a pool of servers. You will assign this pool to virtual servers.
2. Add the addresses of the Oracle HTTP Server hosts to the pool. For example:
 - WEBHOST1:7777
 - WEBHOST2:7777
3. Configure a virtual server in the load balancer for wc.mycompany.com:443.

- For this virtual server, use your system's frontend address as the virtual server address (for example, `wc.mycompany.com`). The frontend address is the externally facing host name used by your system and that will be exposed in the Internet.
 - Configure this virtual server with port 80 and port 443. Any request that goes to port 80 should be redirected to port 443.
 - Specify ANY as the protocol.
 - Enable address and port translation.
 - Enable reset of connections when services and/or nodes are down.
 - Assign the pool created in step 1 to the virtual server.
 - Create rules to filter out access to `/console` and `/em` on this virtual server.
4. Configure a virtual server in the load balancer for `admin.mycompany.com:80`.
- For this virtual server, use your internal administration address as the virtual server address (for example, `admin.mycompany.com`). This address is typically not externalized.
 - Specify HTTP as the protocol.
 - Enable address and port translation.
 - Enable reset of connections when services and/or nodes are down.
 - Assign the pool created in step 1 to the virtual server.
5. Configure a virtual server in the load balancer for `soainternal.mycompany.com:80`.
- For this virtual server, use your internal administration address as the virtual server address (for example, `soainternal.mycompany.com`). This address is typically not externalized.
 - Specify HTTP as the protocol.
 - Enable address and port translation.
 - Enable reset of connections when services and/or nodes are down.
 - Assign the pool created in step 1 to the virtual server.
 - Optionally, create rules to filter out access to `/console` and `/em` on this virtual server.
6. Configure a virtual server in the load balancer for `wcinternal.mycompany.com:80`.
- For this virtual server, use your internal administration address as the virtual server address (for example, `wcinternal.mycompany.com`). This address is typically not externalized.
 - Specify HTTP as the protocol.
 - Enable address and port translation.
 - Enable reset of connections when services and/or nodes are down.
 - Assign the pool created in step 1 to the virtual server.
7. Configure monitors for the Oracle HTTP Server nodes to detect failures in these nodes.
- Set up a monitor to regularly ping the `"/` URL context.

Tip: Use `GET /` instead if the Oracle HTTP Server's document root does not include `index.htm` and Oracle WebLogic Server returns a 404 error for `"/`.

- For the ping interval, specify a value that does not overload your system. You can try 5 seconds as a starting point.
- For the timeout period, specify a value that can account for the longest time response that you can expect from your WebCenter system, that is, specify a value greater than the longest period of time any of your requests to HTTP servers can take.

2.2.3 IPs and Virtual IPs

Configure the Administration Server and the managed servers to listen on different virtual IPs and physical IPs as illustrated in Figure 2-1. As shown in this figure, each VIP and IP is attached to the WebLogic server that uses it. VIP1 is failed manually to restart the Administration Server in SOAHOST2. VIP2 and VIP3 fail over from SOAHOST1 to SOAHOST2 and from SOAHOST2 to SOAHOST1 respectively through Oracle WebLogic Server Migration feature. See *Oracle Fusion Middleware High Availability Guide* for information on the WebLogic Server Migration feature. Physical IPs (non virtual) are fixed to each node. IP1 is the physical IP of SOAHOST1 and is used by the WLS_WSM1 WebServices Policy Manager server. IP2 is the physical IP of SOAHOST2 and is used by the WLS_WSM2 WebServices Policy Manager server.

Figure 2-1 IPs and VIPs Mapped to Administration Server and Managed Servers

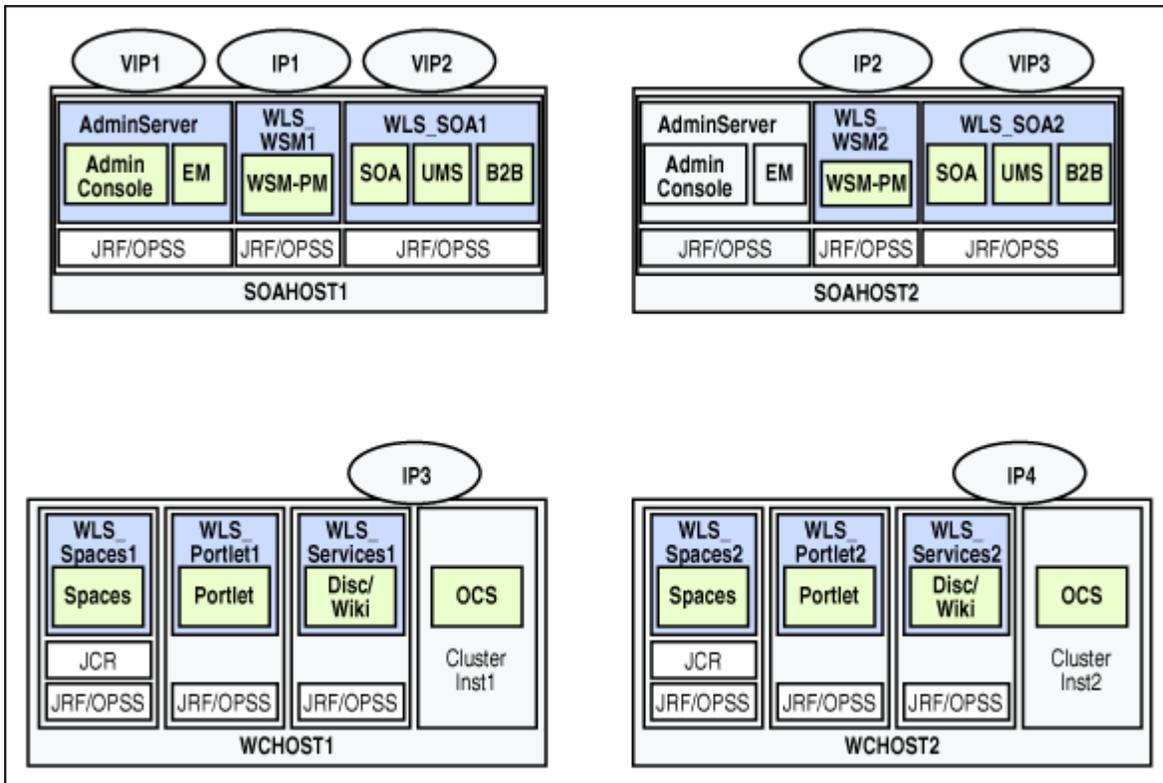


Table 2-2 provides descriptions of the various virtual hosts.

Table 2–2 Virtual Hosts

Virtual IP	VIP Maps to...	Description
VIP1	SOAHOST1VHN1	SOAHOST1VHN1 is the virtual host name that is the listen address for the Administration Server and fails over with manual failover of the Administration Server. It is enabled on the node where the Administration Server process is running (SOAHOST1 by default).
VIP2	SOAHOST1VHN2	SOAHOST1VHN2 is the virtual host name that maps to the listen address for WLS_SOA1 and fails over with server migration of this managed server. It is enabled on the node where WLS_SOA1 process is running (SOAHOST1 by default).
VIP3	SOAHOST2VHN1	SOAHOST2VHN1 is the virtual host name that maps to the listen address for WLS_SOA2 and fails over with server migration of this managed server. It is enabled on the node where WLS_SOA2 process is running (SOAHOST2 by default).

2.2.4 Firewalls and Ports

Many Oracle Fusion Middleware components and services use ports. As an administrator, you must know the port numbers used by these services, and to ensure that the same port number is not used by two services on a host.

Most port numbers are assigned during installation.

[Table 2–3](#) lists the ports used in the WebCenter topology, including the ports that you must open on the firewalls in the topology.

Firewall notation:

- FW0 refers to the outermost firewall.
- FW1 refers to the firewall between the web tier and the application tier.
- FW2 refers to the firewall between the application tier and the data tier.

Table 2–3 Ports Used

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Browser request	FW0	80	HTTP / Load Balancer	Inbound	Timeout depends on all HTML content and the type of process model used for WebCenter.
Browser request	FW0	443	HTTPS / Load Balancer	Inbound	Timeout depends on all HTML content and the type of process model used for WebCenter.
Load balancer to Oracle HTTP Server	n/a	7777	HTTP	n/a	See Section 2.2.2.1, "Configuring the Load Balancer."
OHS registration with Administration Server	FW1	7001	HTTP/t3	Inbound	Set the timeout to a short period (5-10 seconds).
OHS management by Administration Server	FW1	OPMN port (6701) and OHS Admin Port (7779)	TCP and HTTP, respectively	Outbound	Set the timeout to a short period (5-10 seconds).

Table 2–3 (Cont.) Ports Used

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
WSM-PM access	FW1	7010 Range: 7010-7999	HTTP / WLS_ WSM-PM <i>n</i>	Inbound	Set the timeout to 60 seconds.
Communication between WSM Cluster members	n/a	7010	TCP/IP Unicast	n/a	By default, this communication uses the same port as the server's listen address.
Communication between Spaces_ Cluster members	n/a	9000	TCP/IP Unicast	n/a	By default, this communication uses the same port as the server's listen address.
Communication between Portlet_ Cluster members	n/a	9001	TCP/IP Unicast	n/a	By default, this communication uses the same port as the server's listen address.
Communication between Services_ Cluster members	n/a	9002	TCP/IP Unicast	n/a	By default, this communication uses the same port as the server's listen address.
Session replication within a WebLogic Server cluster	n/a	n/a	n/a	n/a	By default, this communication uses the same port as the server's listen address.
Administration Console access	FW1	7001	HTTP / Administration Server and Enterprise Manager t3	Both	You should tune this timeout based on the type of access to the admin console (whether it is planned to use the Oracle WebLogic Server Administration Console from application tier clients or clients external to the application tier).
Node Manager	n/a	5556	TCP/IP	n/a	n/a For actual values, see "Firewalls and Ports" in <i>Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management</i> .
Access Server access	FW1	6021	OAP	Inbound	For actual values, see "Firewalls and Ports" in <i>Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management</i> .
Identity Server access	FW1	6022	OAP	Inbound	
Database access	FW2	1521	SQL*Net	Both	Timeout depends on all database content and on the type of process model used for WebCenter.

Table 2–3 (Cont.) Ports Used

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Oracle Internet Directory access	FW2	389	LDAP	Inbound	You should tune the directory server's parameters based on load balancer, and not the other way around.
Oracle Internet Directory access	FW2	636	LDAP SSL	Inbound	You should tune the directory server's parameters based on load balancer, and not the other way around.
Content Manager access	FW2	9054	TCP/IP socket	n/a	n/a
JOC for OWSM	n/a	9999	TCP/IP	n/a	n/a

2.3 Shared Storage and Recommended Directory Structure

This following section details the directories and directory structure that Oracle recommends for an EDG topology. Other directory layouts are possible and supported, but the model adopted in this guide is chosen for maximum availability, providing both the best isolation of components and symmetry in the configuration and facilitating backup and disaster recovery. The rest of the document uses this directory structure and directory terminology.

This section covers these topics:

- [Section 2.3.1, "Terminology for Directories and Directory Environment Variables"](#)
- [Section 2.3.2, "Recommended Locations for the Different Directories"](#)
- [Section 2.3.3, "Shared Storage Configuration"](#)

2.3.1 Terminology for Directories and Directory Environment Variables

- **ORACLE_BASE:** This environment variable and related directory path refers to the base directory under which Oracle products are installed.
- **MW_HOME:** This environment variable and related directory path refers to the location where Fusion Middleware (FMW) resides.
- **WL_HOME:** This environment variable and related directory path contains installed files necessary to host a WebLogic Server.
- **ORACLE_HOME:** This environment variable and related directory path refers to the location where either Oracle FMW SOA Suite or Oracle WebCenter Suite is installed.
- **ORACLE_COMMON_HOME:** This environment variable and related directory path refers to the Oracle home that contains the binary and library files required for the Oracle Enterprise Manager Fusion Middleware Control and Java Required Files (JRF).
- **DOMAIN Directory:** This directory path refers to the location where the Oracle WebLogic Domain information (configuration artifacts) is stored. Different WLS Servers can use different domain directories even when in the same node as described below.

- **ORACLE_INSTANCE:** An Oracle instance contains one or more system components, such as Oracle Web Cache, Oracle HTTP Server, or Oracle Internet Directory. An Oracle instance directory contains updateable files, such as configuration files, log files, and temporary files.

2.3.2 Recommended Locations for the Different Directories

Oracle Fusion Middleware 11g allows creating multiple SOA or WebCenter managed servers from one single binary installation. This allows the installation of binaries in a single location on a shared storage and the reuse of this installation by the servers in different nodes. However, for maximum availability, Oracle recommends using redundant binary installations. In the EDG model, two MW HOMEs (each of which has a WL_HOME and an ORACLE_HOME for each product suite) are installed in a shared storage. Additional servers (when scaling out or up) of the same type can use either one of these two locations without requiring more installations. Ideally, users should use two different volumes (referred to as VOL1 and VOL2 below) for redundant binary location, thus isolating as much as possible the failures in each volume. For additional protection, Oracle recommends that these volumes are disk mirrored. If multiple volumes are not available, Oracle recommends using mount points to simulate the same mount location in a different directory in the shared storage. Although this does not guarantee the protection that multiple volumes provide, it does allow protection from user deletions and individual file corruption.

When an ORACLE_HOME or a WL_HOME is shared by multiple servers in different nodes, it is recommended to maintain the Oracle Inventory and Middleware home list in those nodes updated for consistency in the installations and application of patches. To update the oraInventory in a node and "attach" an installation in a shared storage to it, use `ORACLE_HOME/oui/bin/attachHome.sh`. To update the Middleware home list to add or remove a WL_HOME, edit the `<user_home>/bea/beahomelist` file. This would be required for any nodes installed additionally to the two ones used in this EDG. An example of the oraInventory and beahomelist updates is provided in the scale-out steps included in this guide.

Oracle recommends also separating the domain directory used by the Administration Server from the domain directory used by managed servers. This allows a symmetric configuration for the domain directories used by managed server, and isolates the failover of the Administration Server. The domain directory for the Administration Server must reside in a shared storage to allow failover to another node with the same configuration. The managed servers' domain directories can reside in a local or shared storage.

It is possible to use a shared domain directory for all managed servers in different nodes or use one domain directory per node. Sharing domain directories for managed servers facilitates the scale-out procedures. In this case, the deployment should conform to the requirements (if any) of the storage system to facilitate multiple machines mounting the same shared volume.

All procedures that apply to multiple local domains apply to a single shared domain. Hence, this enterprise deployment guide uses a model where one domain directory is used per node. The directory can be local or reside in shared storage.

Based on the above assumptions, the following paragraphs describe the directories recommended. Wherever a shared storage location is directly specified, it is implied that shared storage is required for that directory. When using local disk or shared storage is optional the mount specification is qualified with "if using a shared disk." The shared storage locations are examples and can be changed as long as the provided mount points are used. However, Oracle recommends this structure in the shared storage device for consistency and simplicity.

ORACLE_BASE:

/u01/app/oracle

MW_HOME (application tier):

ORACLE_BASE/product/fmw

- Mount point: ORACLE_BASE/product/fmw
- Shared storage location: ORACLE_BASE/product/fmw (VOL1 and VOL2)

MW_HOME (web tier):

ORACLE_BASE/product/fmw/web

- Mount point: ORACLE_BASE/product/fmw
- Shared storage location: ORACLE_BASE/product/fmw (VOL1 and VOL2)

WL_HOME:

MW_HOME/wlserver_10.3

ORACLE_HOME:

MW_HOME/wc

ORACLE_COMMON_HOME:

MW_HOME/oracle_common

ORACLE_INSTANCE:

ORACLE_BASE/admin/<instance_name>

- If you are using a shared disk, the mount point on the machine is ORACLE_BASE/admin/<instance_name> mounted to ORACLE_BASE/admin/<instance_name> (VOL1).

Note: (VOL1) is optional; you could also use (VOL2).

Domain Directory for Administration Server Domain Directory:

ORACLE_BASE/admin/<domain_name>/aserver/<domain_name> (The last "domain_name" is added by config wizard)

- Mount point on machine: ORACLE_BASE/admin/<domain_name>/aserver
- Shared storage location: ORACLE_BASE/admin/<domain_name>/aserver

Domain Directory for Managed Server Domain Directory:

ORACLE_BASE/admin/<domain_name>/mserver/<domain_name>

- If you are using a shared disk, the mount point on the machine is ORACLE_BASE/admin/<domain_name>/mserver mounted to /ORACLE_BASE/admin/<domain_name>/Noden/mserver/ (each node uses a different domain directory for managed servers).

Note: This procedure is really shared storage dependent. The above example is specific to NAS, but other storage types may provide this redundancy with different types of mappings.

Location for JMS file-based stores and Tlogs (SOA only):

ORACLE_BASE/admin/<domain_name>/<soa_cluster_name>/jms

ORACLE_BASE/admin/<domain_name>/<soa_cluster_name>/tlogs

- Mount point: ORACLE_BASE/admin/<domain_name>/<soa_cluster_name>/
- Shared storage location: ORACLE_BASE/admin/<domain_name>/<soa_cluster_name>/

Location for Application Directory:

ORACLE_BASE/admin/<domain_name>/apps

- Mount point: ORACLE_BASE/admin/<domain_name>/apps
- Shared storage location: ORACLE_BASE/admin/<domain_name>/aserver

Figure 2-2 shows this directory structure in a diagram.

Figure 2-2 Directory Structure

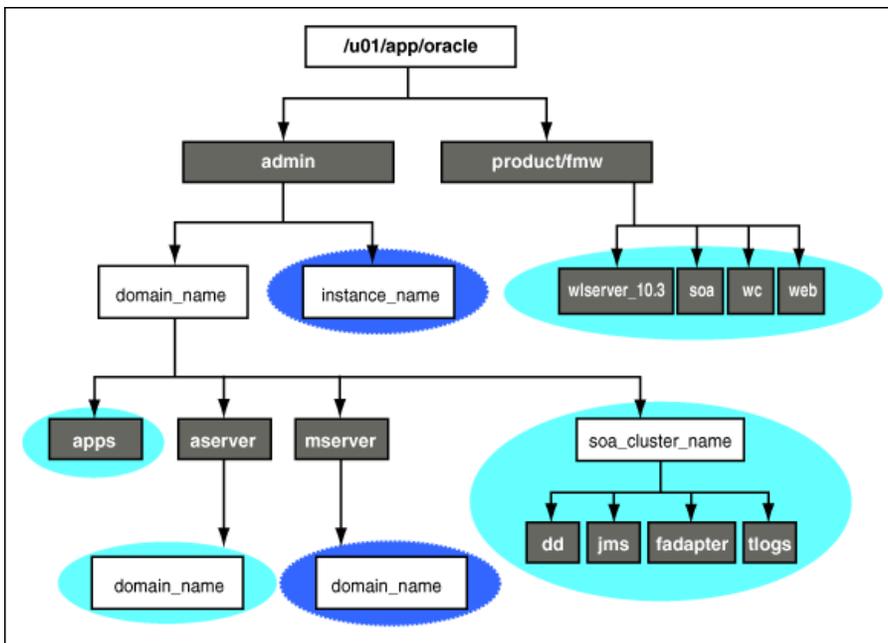


Diagram representation of the directory structure that is described in the text preceding this image.

The directory structure in Figure 2-2 does not show other required internal directories, such as oracle_common and jrockit.

Table 2-4 explains what the various color-coded elements in the diagram mean.

Table 2-4 Directory Structure Elements

Element	Explanation
	The Administration Server domain directories, applications, deployment plans, file adapter control directory, JMS and TX logs, and the entire MW_HOME are on a shared disk.

Table 2-4 (Cont.) Directory Structure Elements

Element	Explanation
	The managed server domain directories can be on a local disk or a shared disk. Further, if you want to share the managed server domain directories on multiple nodes, then you must mount the same shared disk location across the nodes. The instance_name directory for the web tier can be on a local disk or a shared disk.
	Fixed name.
	Installation-dependent name.

Figure 2-3 shows an example configuration for shared storage with multiple volumes for WebCenter.

Figure 2-3 Example Configuration for Shared Storage

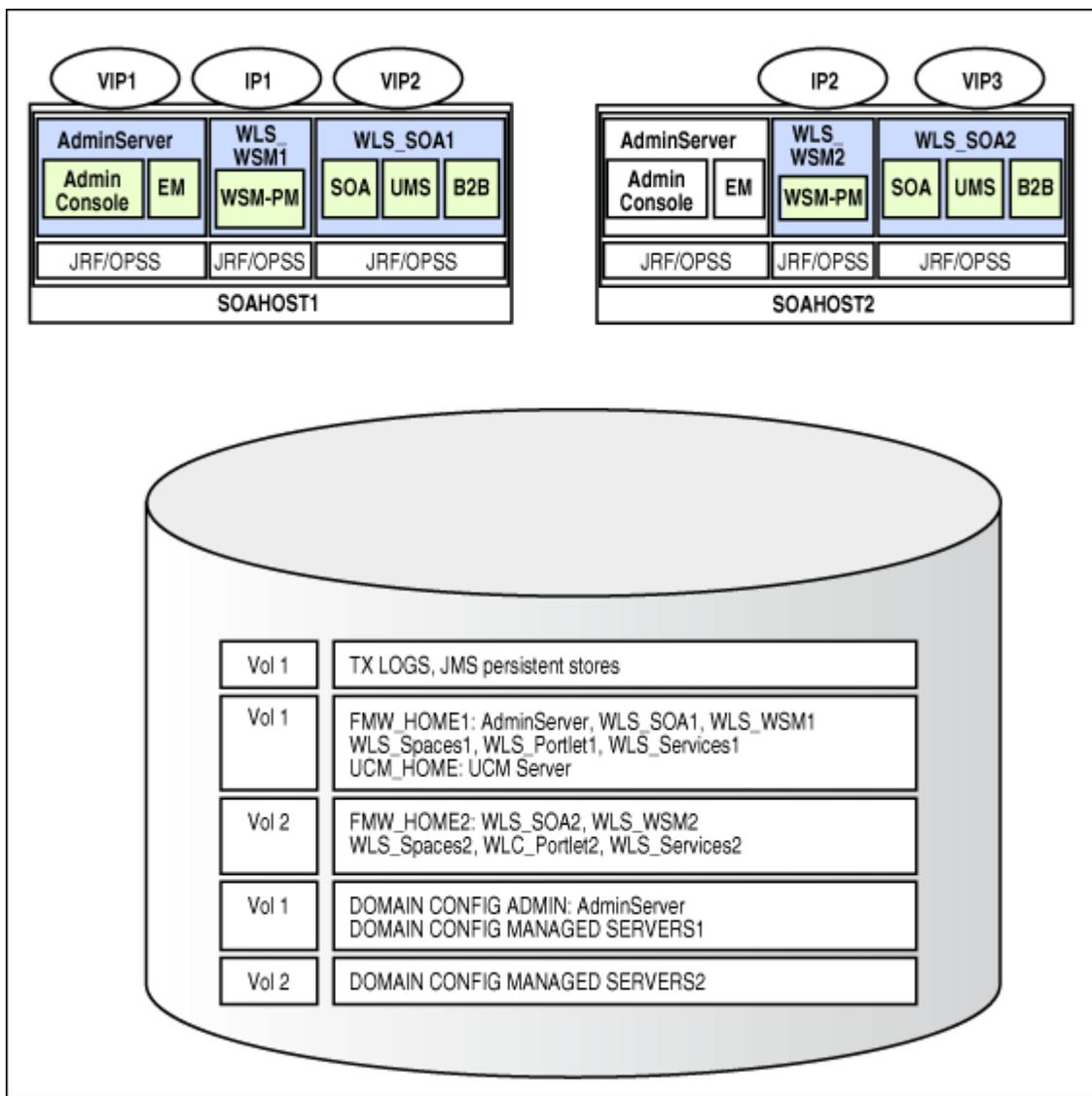


Table 2–5 summarizes the directory structure for the domain. In the table, "WLS_WC" refers to all the WebCenter managed servers: WLS_Spaces, WLS_Portlet and WLS_Services.

Table 2–5 Contents of Shared Storage

Server	Type of Data	Volume in Shared Storage	Directory	Files
WLS_SOA1	Tx Logs	VOL1	ORACLE_BASE/admin/<domain_name>/<soa_cluster_name>/tlogs	The transaction directory is common (decided by WebLogic Server), but the files are separate.
WLS_SOA2	Tx Logs	VOL1	ORACLE_BASE/admin/<domain_name>/<soa_cluster_name>/tlogs	The transaction directory is common (decided by WebLogic Server), but the files are separate.
WLS_SOA1	JMS Stores	VOL1	ORACLE_BASE/admin/<domain_name>/<soa_cluster_name>/jms	The transaction directory is common (decided by WebLogic Server), but the files are separate; for example: SOAJMSStore1, UMSJMSStore1, and so on.
WLS_SOA2	JMS Stores	VOL1	ORACLE_BASE/admin/<domain_name>/<soa_cluster_name>/jms	The transaction directory is common (decided by WebLogic Server), but the files are separate; for example: SOAJMSStore2, UMSJMSStore2, etc.
WLS_SOA1	WLS Install	VOL1	MW_HOME	Individual in each volume, but both servers see same directory structure.
WLS_SOA2	WLS Install	VOL2	MW_HOME	Individual in each volume, but both servers see same directory structure.
WLS_WC1	WLS Install	VOL1	MW_HOME	Individual in each volume, but both servers see same directory structure.
WLS_WC2	WLS Install	VOL2	MW_HOME	Individual in each volume, but both servers see same directory structure.
WLS_SOA1	SOA Install	VOL1	MW_HOME/ soa	Individual in each volume, but both servers see same directory structure.
WLS_SOA2	SOA Install	VOL2	MW_HOME/ soa	Individual in each volume, but both servers see same directory structure.
WLS_WC1	WebCenter Install	VOL1	MW_HOME/wc	Individual in each volume, but both servers see same directory structure.
WLS_WC2	WebCenter Install	VOL2	MW_HOME/wc	Individual in each volume, but both servers see same directory structure.
WLS_SOA1	Domain Config	VOL1	ORACLE_BASE/admin/<domain_name>/msserver/<domain_name>	Individual in each volume, but both servers see same directory structure.

Table 2–5 (Cont.) Contents of Shared Storage

Server	Type of Data	Volume in Shared Storage	Directory	Files
WLS_SOA2	Domain Config	VOL2	ORACLE_BASE/admin/<domain_name>/msserver/<domain_name>	Individual in each volume, but both servers see same directory structure.
WLS_WC1	Domain Config	VOL1	MW_HOME/user_projects_domains/edgdomain	Individual in each volume, but both servers see same directory structure.
WLS_WC2	Domain Config	VOL2	MW_HOME/user_projects_domains/edgdomain	Individual in each volume, but both servers see same directory structure.

2.3.3 Shared Storage Configuration

The following steps show to create and mount shared storage locations so that SOAHOST1, SOAHOST2, WCHOST1, and WCHOST2 can see the same location for binary installation in two separate volumes.

"nasfiler" is the shared storage filer.

From SOAHOST1 and WCHOST1:

```
SOAHOST1> mount nasfiler:/vol/vol1/u01/app/oracle/product/fmw
/u01/app/oracle/product/fmw -t nfs
```

From SOAHOST2 and WCHOST2:

```
SOAHOST2> mount nasfiler:/vol/vol2/u01/app/oracle/product/fmw
/u01/app/oracle/product/fmw -t nfs
```

If only one volume is available, users can provide redundancy for the binaries by using two different directories in the shared storage and mounting them to the same dir in the SOA Servers:

From SOAHOST1:

```
SOAHOST1> mount nasfiler:/vol/vol1/u01/app/oracle/product/fmw1
/u01/app/oracle/product/fmw -t nfs
```

From SOAHOST2:

```
SOAHOST2> mount nasfiler:/vol/vol2/u01/app/oracle/product/fmw2
/u01/app/oracle/product/fmw -t nfs
```

The following commands show how to share the SOA TX logs location across different nodes:

```
SOAHOST1> mount nasfiler:/vol/vol1/u01/app/oracle/stores/soadomain/soa_
cluster/tlogs
/u01/app/oracle/stores/soadomain/soa_cluster/tlogs -t nfs
```

```
SOAHOST2> nasfiler:/vol/vol1/u01/app/oracle/stores/soadomain/soa_cluster/tlogs
/u01/app/oracle/stores/soadomain/soa_cluster/tlogs -t nfs
```

Note: The shared storage can be a NAS or SAN device. The following illustrates an example of creating storage for a NAS device from SOAHOST1. The options may differ.

```
SOAHOST1> mount nasfiler:/vol/vol1/fmw11shared ORACLE_BASE/wls -t
nfs -o rw,bg,hard,nointr,tcp,vers=3,timeo=300,rsize=32768,
wsize=32768
```

Contact your storage vendor and machine administrator for the correct options for your environment.

2.4 LDAP as Credential and Policy Store

With Oracle Fusion Middleware, you can use different types of credential and policy stores in a WebLogic domain. Domains can use stores based on XML files or on different types of LDAP providers. When a domain uses an LDAP store, all policy and credential data is kept and maintained in a centralized store. However, when using XML policy stores, the changes made on managed servers are not propagated to the Administration Server unless they use the same domain home.

An Oracle Fusion Middleware SOA Suite Enterprise Deployment Topology uses different domain homes for the Administration Server and the managed server as described in the [Section 2.3, "Shared Storage and Recommended Directory Structure."](#) Derived from this, and for integrity and consistency purposes, Oracle requires the use of an LDAP as policy and credential store in context of Oracle Fusion Middleware SOA Suite Enterprise Deployment Topology. To configure the Oracle Fusion Middleware SOA Suite Enterprise Deployment Topology with an LDAP as Credential and Policy store, follow the steps in [Section 10.1, "Credential and Policy Store Configuration."](#)

Installing Oracle HTTP Server

You install and configure Oracle HTTP Server on nodes in the web tier.

This chapter contains the following sections:

- [Section 3.1, "Installing Oracle HTTP Server on WEBHOST1 and WEBHOST2"](#)
- [Section 3.2, "Validating Oracle HTTP Server Through the Load Balancer"](#)
- [Section 3.3, "Backing Up Oracle HTTP Server"](#)

3.1 Installing Oracle HTTP Server on WEBHOST1 and WEBHOST2

Follow these steps to install Oracle HTTP Server on WEBHOST1 and WEBHOST2:

1. Check that your machines meet the following requirements:
 - Ensure that the system, patch, kernel, and other requirements are met as specified in the installation guide.
 - Because Oracle HTTP Server is installed by default on port 7777, you must ensure that port 7777 is not used by any service on the nodes. To check if this port is in use, run the following command before installing Oracle HTTP Server. You must free the ports if they are in use.

```
netstat -an | grep 7777
```
2. On Linux platforms, if the `/etc/oraInst.loc` file exists, check that its contents are correct. Specifically, check that the inventory directory is correct and that you have write permissions for that directory.

If the `/etc/oraInst.loc` file does not exist, you can skip this step.
3. Start the Oracle Universal Installer from the Oracle Fusion Middleware 11g WebTier and Utilities DVD by issuing this command:

```
runInstaller
```
4. If the Specify Inventory Directory screen appears, enter the location for the inventory and the user group and then click **OK**; otherwise, ignore steps 4 and 5 and continue with step 6.
5. Execute the root privileged actions as indicated in the dialog and then click **OK**.
6. In the **Welcome** screen, click **Next**.
7. In the **Select Installation Type** screen, select **Install and Configure**, and click **Next**.
8. In the **Prerequisite Checks** screen, ensure that all the prerequisites are met, then click **Next**.

9. In the **Specify Installation Location** screen, do the following:

- On WEBHOST1, set the **Location** to:

`MW_HOME/web`

- On WEBHOST2, set the **Location** to:

`MW_HOME/web`

Click **Next**.

10. In the **Configure Components** screen, do the following:

- Select **Oracle Http Server**.
- Do *not* select **Oracle Web Cache**.
- Do *not* select **Associate Selected Components with WebLogic Domain** because you have not yet installed WebLogic Server.

Click **Next**.

11. In the **Specify Component Details** screen, do the following:

- Enter the following values for WEBHOST1:
 - **Instance Home Location:** `ORACLE_BASE/admin/<instance_name>`
 - **Instance Name:** `ohs_instance1`
 - **OHS Component Name:** `ohs1`
- Enter the following values for WEBHOST2:
 - **Instance Home Location:** `ORACLE_BASE/admin/<instance_name>`
 - **Instance Name:** `ohs_instance2`
 - **OHS Component Name:** `ohs2`

Click **Next**.

12. In the **Configure Ports** screen, do the following:

- Select **Specify Ports using Configuration File** and copy the *staticports.ini* template file from your installation disk (the file is located in the `/Disk1/stage/Response` directory) to your user's home. Then use the **Browse** button to select this file.
- Click **View/Edit File** to open the *staticports.ini* file in an editor.
- Change the Oracle HTTP Server port in that file to 7777.
- Save the file.

Click **Next**.

Note: For more information on setting ports, refer to *Oracle Fusion Middleware Installation Guide for Oracle SOA Suite*.

13. In the **Specify Security Updates** screen, enter your e-mail address to receive e-mail notifications of security issues (if required). Enter your Oracle Support Password to receive security updates through My Oracle Support.

14. In the **Installation Summary** screen, ensure that the selections are correct, and click **Install**.

15. In the **Configuration** screen, multiple configuration assistants are launched in succession, which can be a lengthy process. When it completes, the **Configuration Completed** screen appears.
16. In the **Installation Completed** screen, click **Finish** to exit.

3.2 Validating Oracle HTTP Server Through the Load Balancer

Define the directives of the <VirtualHost> section of the httpd.conf file on both OHS servers. This file is located in the ORACLE_BASE/admin/<instance_name>/config/OHS/ohs1 (or ohs2) directory. Add the following entries to the file:

```
NameVirtualHost *:7777
<VirtualHost *:7777>
    ServerName https://wc.mycompany.com:443
    ServerAdmin you@your.address
    RewriteEngine On
    RewriteOptions inherit
</VirtualHost>

NameVirtualHost *:7777
<VirtualHost *:7777>
    ServerName admin.mycompany.com:80
    ServerAdmin you@your.address
    RewriteEngine On
    RewriteOptions inherit
</VirtualHost>

NameVirtualHost *:7777
<VirtualHost *:7777>
    ServerName wcinternal.mycompany.com:80
    ServerAdmin you@your.address
    RewriteEngine On
    RewriteOptions inherit
</VirtualHost>
```

Make sure that you restart both OHS servers after modifying the httpd.conf files:

```
WEBHOST> cd ORACLE_BASE/admin/<instance_name>/bin
WEBHOST> opmnctl start
WEBHOST> opmnctl restartproc process-type=OHS
```

Verify that the server status is reported as "Running" in the Admin Console. If the server is shown as "Starting" or "Resuming," wait for the server status to change to "Started." If another status is reported (such as "Admin" or "Failed"), check the server output log files for errors. See [Section 11.5, "Troubleshooting"](#) for possible causes.

Access the following URLs to ensure that your load balancer and Oracle HTTP Server are configured properly:

- <http://wc.mycompany.com/index.html>
- <http://admin.mycompany.com/index.html>
- <http://wcinternal.mycompany.com/index.html>
- <https://wc.mycompany.com/index.html>

3.3 Backing Up Oracle HTTP Server

After you have verified that the Oracle HTTP Server installation is working, back up the installation. This is a quick backup for the express purpose of immediate restore in case of problems in the further steps. The backup destination is the local disk. This backup can be discarded once the enterprise deployment setup is complete. At this point, the regular deployment-specific backup and recovery process can be initiated. The *Oracle Fusion Middleware Administrator's Guide* provides further details. For information on describing the Oracle HTTP Server data that must be backed up and restored, refer to the "Backup and Recovery Recommendations for Oracle HTTP Server" section in this guide. For information on how to recover components, see "Recovery of Components" and "Recovery After Loss of Component" sections in the guide. For recommendations specific to recovering from the loss of a host, see the "Recovering Oracle HTTP Server to a Different Host" in the guide.

To back up the installation at this point, complete these steps:

1. Shut down the instance using `opmnctl`.

```
ORACLE_BASE/admin/<instance_name>/bin/opmnctl stopall
```

2. Back up the Middleware Home on the web tier using the following command:

```
tar -cvpf BACKUP_LOCATION/web.tar $MW_HOME
```

3. Back up the Instance Home on the web tier using the following command:

```
tar -cvpf BACKUP_LOCATION/web_<instance_name>.tar $ORACLE_INSTANCE
```

4. Start the instance using `opmnctl`:

```
ORACLE_BASE/admin/<instance_name>/bin/opmnctl startall
```

Creating a Domain

This chapter describes how to create a domain using the Configuration Wizard, Oracle WebLogic Server Administration Console, Oracle Enterprise Manager, and Oracle WSM Policy Manager. You can extend the domain to add WebCenter components.

Important: Oracle strongly recommends that you read the release notes for any additional installation and deployment considerations prior to starting the setup process.

This chapter contains the following sections:

- [Section 4.1, "Installing Oracle Fusion Middleware Home"](#)
- [Section 4.2, "Backing Up the Installation"](#)
- [Section 4.3, "Enabling VIP1 in SOAHOST1"](#)
- [Section 4.4, "Running the Configuration Wizard on SOAHOST1 to Create a Domain"](#)
- [Section 4.5, "Creating a Separate Domain Directory for Managed Servers in the Same Node as the Administration Server"](#)
- [Section 4.6, "Creating boot.properties for the Administration Server on SOAHOST1"](#)
- [Section 4.7, "Starting the Administration Server on SOAHOST1"](#)
- [Section 4.8, "Validating the Administration Server"](#)
- [Section 4.9, "Applying the Java Required Files \(JRF\) Template to the WSM-PM_Cluster"](#)
- [Section 4.10, "Disabling Host Name Verification for the Oracle WebLogic Administration Server and the WLS_WSM1 Managed Server"](#)
- [Section 4.11, "Starting Node Manager on SOAHOST1"](#)
- [Section 4.12, "Starting and Validating the WLS_WSM1 Managed Server"](#)
- [Section 4.13, "Propagating the Domain Configuration to SOAHOST2 Using the unpack Utility"](#)
- [Section 4.14, "Disabling Host Name Verification for the WLS_WSM2 Managed Server"](#)
- [Section 4.15, "Starting Node Manager on SOAHOST2"](#)
- [Section 4.16, "Starting and Validating the WLS_WSM2 Managed Server"](#)

- [Section 4.17, "Configuring the Java Object Cache for Oracle WSM"](#)
- [Section 4.18, "Configuring Oracle HTTP Server for the Administration Server and the WLS_WSM-PMn Managed Servers"](#)
- [Section 4.19, "Registering Oracle HTTP Server With WebLogic Server"](#)
- [Section 4.20, "Setting the Frontend URL for the Administration Console and Setting Redirection Preferences"](#)
- [Section 4.21, "Validating Access Through Oracle HTTP Server"](#)
- [Section 4.22, "Manually Failing Over the Administration Server to SOAHOST2"](#)
- [Section 4.23, "Validating Access to SOAHOST2 Through Oracle HTTP Server"](#)
- [Section 4.24, "Failing the Administration Server Back to SOAHOST1"](#)
- [Section 4.25, "Backing Up the Installation"](#)

4.1 Installing Oracle Fusion Middleware Home

As described in [Section 2.3, "Shared Storage and Recommended Directory Structure,"](#) you install Oracle Fusion Middleware in at least two storage locations for redundancy.

You must install the following components of Oracle Fusion Middleware:

- Oracle WebLogic Server (see [Section 4.1.1, "Installing Oracle WebLogic Server"](#))
- Oracle WebCenter (see [Section 4.1.2, "Installing Oracle Fusion Middleware for WebCenter"](#))

4.1.1 Installing Oracle WebLogic Server

Perform these steps to install Oracle WebLogic Server on SOAHOST1, SOAHOST2, WCHOST1, and WCHOST2.

1. Start the Oracle WebLogic Server installer.
2. In the Welcome screen, click **Next**.
3. In the Choose Middleware Home Directory screen, do the following:
 - Select **Create a New Middleware Home**.
 - For Middleware Home Directory, enter **MW_HOME**.

Note: ORACLE_BASE is the base directory under which Oracle products are installed. The recommended value is /u01/app/oracle. See [Section 2.3, "Shared Storage and Recommended Directory Structure"](#) for more information.

Click **Next**.

4. In the Register for Security Updates screen, enter your contact information so that you can be notified of security updates, and click **Next**.
5. In the Choose Install Type screen, select **Custom**, and click **Next**.
6. In the Choose Products and Components screen, click **Next**.
7. In the JDK Selection screen, select *only* **Oracle JRockit 1.6.0_05 SDK**, and click **Next**.

8. In the Choose Product Installation Directories screen, accept the directory **WL_HOME**, and click **Next**.
9. In the Installation Summary screen, click **Next**.
10. In the Installation Complete screen, unselect **Run QuickStart**, and click **Done**.

4.1.2 Installing Oracle Fusion Middleware for WebCenter

Perform these steps to install Oracle Fusion Middleware for WebCenter on SOAHOST1, SOAHOST2, WCHOST1, and WCHOST2.

1. On Linux platforms, if the `/etc/oraInst.loc` file exists, check that its contents are correct. Specifically, check that the inventory directory is correct and that you have write permissions for that directory.

If the `/etc/oraInst.loc` file does not exist, you can skip this step.

2. Start the installer for Oracle Fusion Middleware for WebCenter.

```
SOAHOST1> runInstaller
```

When the installer prompts you for a JRE/JDK location, enter the Oracle SDK location created in the Oracle WebLogic Server installation, for example, **MW_HOME/jrockit_160_14_R27.6.4-18**.

3. In the Specify Inventory Directory screen, do the following:
 - a. Enter `HOME/oraInventory`, where `HOME` is the home directory of the user performing the installation (this is the recommended location).
 - b. Enter the OS group for the user performing the installation.
 - c. Click **Next**.

Follow the instructions on screen to execute `/createCentralInventory.sh` as root. Click **OK**.

4. In the Welcome screen, click **Next**.
5. In the Prerequisite Check screen, verify that the checks complete successfully, and click **Next**.
6. Specify the installation location. Select the previously installed Middleware Home from the drop-down list. For the Oracle Home directory, enter the directory name (**wc**).

Click **Next**.

7. In the Specify UCM Configuration Options screen, clear the "Install and Configure Universal Content Management" check box. You will install Oracle Content Server (OCS) separately at a later stage.

Click **Next**.

8. In the Installation Summary screen, click **Install**.
9. In the Installation Complete screen, click **Finish**.

4.2 Backing Up the Installation

The Fusion Middleware Home should be backed up now (make sure that you stop the server first):

```
SOAHOST1> tar -cvpf fmhomeback.tar MW_HOME
```

This creates a backup of the installation files for both Oracle WebLogic Server and Oracle Fusion Middleware for WebCenter.

4.3 Enabling VIP1 in SOAHOST1

Please note that this step is required for failover of the Administration Server, regardless of whether SOA is installed or not.

You are associating the Administration Server with a virtual hostname (SOAHOST1VHN1). This Virtual Host Name must be mapped to the appropriate VIP (VIP1) either by a DNS Server or by a custom `/etc/host` entry. Check that SOAHOST1VHN1 is available per your name resolution system, (DNS server, `/etc/hosts`), in the required nodes in your SOA topology. The VIP (VIP1) that is associated to this Virtual Host Name (SOAHOST1VHN1) must be enabled in SOAHOST1.

To enable the virtual IP on Linux, run the `ifconfig` command as root:

```
/sbin/ifconfig <interface:index> <IPAddress> netmask <netmask>  
/sbin/arping -q -U -c 3 -I <interface> <IPAddress>
```

For example:

```
/sbin/ifconfig ethX:Y 100.200.140.206 netmask 255.255.255.0
```

Enable your network to register the new location of the virtual IP, for example:

```
/sbin/arping -q -U -c 3 -I ethX 100.200.140.206
```

Validate that the address is available by pinging it from another node, for example:

```
/bin/ping 100.200.140.206
```

In this example 'ethX' is the ethernet interface (eth0 or eth1) and Y is the index (0, 1, 2, etc.).

4.4 Running the Configuration Wizard on SOAHOST1 to Create a Domain

Run the Configuration Wizard from the WebCenter home directory to create a domain containing the Administration Server and Oracle Web Services Manager. Later, you will extend the domain to contain WebCenter components.

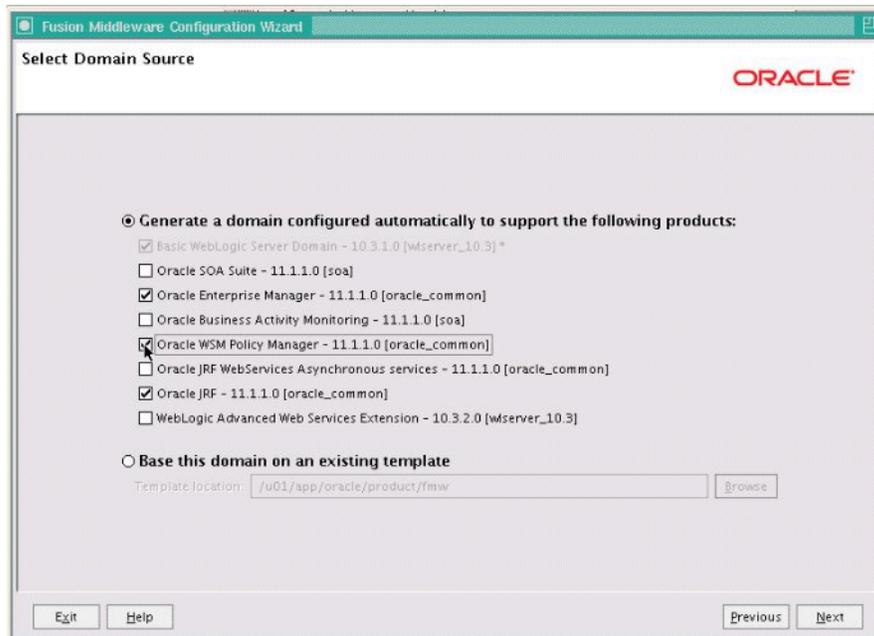
1. Ensure that the database where you installed the repository is running. For RAC databases, it is recommended that all instances are running, so that the validation check later on becomes more reliable.
2. Change directory to the location of the Configuration Wizard. This is within the WebCenter home directory.

```
SOAHOST1> cd MW_HOME/wc/common/bin
```

3. Start the Oracle Fusion Middleware Configuration Wizard:

```
SOAHOST1> ./config.sh
```

4. In the Welcome screen, select **Create a New WebLogic Domain**, and click **Next**.
5. The Select Domain Source screen is displayed (Figure 4-1).

Figure 4–1 Select Domain Source Screen

Screenshot of the Select Domain Source screen, where you select the products that the newly created WebLogic domain will support automatically. It is described in further detail in the text following this image.

In the Select Domain Source screen, do the following:

- Select **Generate a domain configured automatically to support the following products**.
- Select the following products:
 - **Basic WebLogic Server Domain - 10.3.1.0 [wls_server_10.3][wc]** (this should be selected automatically)
 - **Oracle Enterprise Manager - 11.1.1.0 [oracle_common]**
 - **Oracle WSM Policy Manager 11.1.1.0 [oracle_common]**
 - **Oracle JRF - 11.1.1.0 [oracle_common]** (this should be selected automatically)

If you accidentally deselect some of the targets, make sure that the following selections are made in this screen:

- Oracle Enterprise Manager
- Oracle WSM Policy Manager
- Oracle JRF

Click **Next**.

Note: If multiple Oracle Homes are installed (for example a WebCenter Home and a SOA Home), available products will show up for both homes. In this step, select only products from the WebCenter home (wc). This is indicated by brackets at the end of the product name; for example, "Oracle JRF - 11.1.1.0 [wc]."

6. In the Specify Domain Name and Location screen, enter the domain name (wcedg_domain).

Make sure that the domain directory matches the directory and shared storage mount point recommended in [Chapter 2, "Database and Environment Preconfiguration"](#): enter ORACLE_BASE/admin/<domain_name>/aserver/ for the domain directory and ORACLE_BASE/admin/<domain_name>/apps for the application directory. This directory should be in shared storage.
7. Click **Next**.
8. In the Configure Administrator Username and Password screen, enter the username and password to be used for the domain's administrator.

Click **Next**.
9. In the Configure Server Start Mode and JDK screen, do the following:
 - For WebLogic Domain Startup Mode, select **Production Mode**.
 - For JDK Selection, select **Oracle JRockit 1.6.0_14 SDK**.Click **Next**.
10. In the Configure JDBC Components Schema screen, do the following:
 - a. Select the OWSM MDS schema.
 - b. Select **Configure selected component schemas as RAC multi data source schemas in the next panel**.
 - c. Click **Next**.
11. The Configure RAC Multi Data Sources Component Schema screen is displayed ([Figure 4-2](#)).

Figure 4–2 Configure RAC Multi Data Source Component Schema Screen

Note: Change only the input fields below that you wish to modify and values will be applied to all selected rows.

Driver: *Oracle's Driver (Thin) for RAC Service-Instance c[...]

Service Name: wcedg.mycompany.com

Username: wcedg_mds

Password: *****

Host Name	Instance Name	Port
custdbhost1-vip.mycom	wcedgdb1	1521
custdbhost2-vip.mycom	wcedgdb2	1521

Add Delete

Multi Data Source Schema	Service Name	Schema Owner	Schema Password
<input checked="" type="checkbox"/> OWSM MDS Schema	wcedg.mycompany.com	wcedg_mds	*****

Exit Help Previous Next

In this screen, do the following:

- a. Enter values for the following fields, specifying the connect information for the RAC database that was seeded with RCU.
 - **Driver:** Select **Oracle driver (Thin) for RAC Service-Instance connections, Versions:10, 11.**
 - **Service Name:** Enter the service name of the database, for example, `wcedg.mycompany.com`.
 - **Username:** Enter the complete user name (including the prefix) for the schemas.
 - **Password:** Enter the password to use to access the schemas.
 - b. Enter the host name, instance name, and port.
 - c. Click **Add**.
 - d. Repeat this for each RAC instance.
 - e. Click **Next**.
12. In the Test JDBC Data Sources screen, the connections should be tested automatically. The **Status** column displays the results. Ensure that all connections were successful. If not, click **Previous** to return to the previous screen and correct your entries.

Click **Next** when all the connections are successful.

13. In the Select Advanced Configuration screen, select the following:
 - **Administration Server**

- **Managed Servers, Clusters and Machines**
- **Deployment and Services**

Click **Next**.

14. In the Configure the Administration Server screen, enter the following values:

- Name: **AdminServer**
- Listen Address: enter SOAHOST1VHN1.
- Listen Port: **7001**
- SSL listen port: **N/A**
- SSL enabled: **unchecked**

Click **Next**.

15. In the Configure Managed Servers screen, click **Add** to add the following managed servers:

Table 4–1 Managed Servers

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled
WLS_WSM1	SOAHOST1	7010	n/a	No
WLS_WSM2	SOAHOST2	7010	n/a	No

Click **Next**.

16. In the Configure Clusters screen, Click **Add** to add the following clusters:

Table 4–2 Clusters

Name	Cluster Messaging Mode	Multicast Address	Multicast Port	Cluster Address
WSM-PM_Cluster	unicast	n/a	n/a	Leave it empty.

Click **Next**.

17. In the Assign Servers to Clusters screen, assign servers to clusters as follows:

- **WSM-PM_Cluster:**
 - WLS_WSM1
 - WLS_WSM2

Click **Next**.

18. In the Configure Machines screen, do the following:

- Click the **Unix Machine** tab and then click **Add** to add the following machines:

Table 4–3 Machines

Name	Node Manager Listen Address
SOAHOST1	SOAHOST1
SOAHOST2	SOAHOST2

Leave all other fields to their default values.

Click **Next**.

19. In the **Assign Servers to Machines** screen, assign servers to machines as follows:

- **SOAHOST1:**
 - AdminServer
 - WLS_WSM1
- **SOAHOST2:**
 - WLS_WSM2

Click **Next**.

20. In the **Target Deployments to Clusters or Servers** screen, make sure that the **wsm-pm** application and the **oracle.wsm.seedpolicies** library is targeted to the **WSM-PM_Cluster** only. Make sure that all other deployments are targeted to the **AdminServer**. Click **Next**.

21. In the **Target Services to Clusters or Servers** screen, select the following:

- On the left, select **WSM-PM_Cluster**. On the right, select **JDBC System Resource** (this automatically selects all the wsm datasources (mds-owsm)).
- On the left, select **Admin Server**. On the right, select **JDBC System Resource** (this automatically selects all the wsm datasources (mds-owsm)).

All JDBC system resources should be targeted to both the Admin Server and WSM-PM_Cluster.

- On the left, select **WSM-PM_Cluster**. On the right, select **JOC-Shutdown**, **JOC-Startup**, and **OWSM Startup Class**.
- On the left, select **Admin Server**. On the right, deselect **JOC-Shutdown** and **JOC-Startup**. Make sure these services are not targeted to the Admin Server.

JOC-Shutdown, **JOC-Startup**, and **OWSM Startup Class** should be targeted only to the **WSM-PM_Cluster**.

- Make sure that all the remaining services are targeted to the **Admin Server**.

22. In the **Configuration Summary** screen, click **Create**.

23. In the **Create Domain** screen, click **Done**.

4.5 Creating a Separate Domain Directory for Managed Servers in the Same Node as the Administration Server

Use the `pack` and `unpack` commands to separate the domain directory used by the Administration Server from the domain directory used by the managed server in SOAHOST1 as recommended in [Chapter 2, "Database and Environment Preconfiguration."](#)

1. Run the `pack` command on SOAHOST1 to create a template pack as follows:

```
SOAHOST1> cd MW_HOME/wc/common/bin
```

```
SOAHOST1> ./pack.sh -managed=true -domain=ORACLE_BASE/admin/<domain_name>/aserver/<domain_name> -template=wcdomaintemplate.jar -template_name=wc_domain_template
```

2. Run the unpack command on SOAHOST1 to unpack the template in the managed server domain directory as follows:

```
SOAHOST1> cd MW_HOME/wc/common/bin
```

```
SOAHOST1> ./unpack.sh -domain=ORACLE_BASE/admin/<domain_name>/msserver/<domain_name> -template=wcdomaintemplate.jar
```

4.6 Creating boot.properties for the Administration Server on SOAHOST1

Create a `boot.properties` file for the Administration Server on SOAHOST1. The `boot.properties` file enables the Administration Server to start without prompting you for the administrator username and password.

1. Create the following directory structure:

```
mkdir ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>/servers
```

```
mkdir ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>/servers/AdminServer
```

```
mkdir ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>/servers/AdminServer/security
```

2. In a text editor, create a file called `boot.properties` in the last directory created in the previous step, and enter the following lines in the file:

```
username=<adminuser>
password=<password>
```

Note: When you start the Administration Server, the username and password entries in the file get encrypted. You start the Administration Server in [Section 4.7, "Starting the Administration Server on SOAHOST1."](#)

For security reasons, you want to minimize the time the entries in the file are left unencrypted: after you edit the file, you should start the server as soon as possible so that the entries get encrypted.

4.7 Starting the Administration Server on SOAHOST1

Start the Administration Server:

```
SOAHOST1> cd ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>/bin
SOAHOST1> ./startWebLogic.sh
```

4.8 Validating the Administration Server

Perform these steps to ensure that the Administration Server is properly configured:

1. In a browser, go to `http://SOAHOST1VHN1:7001/console`.
2. Log in as the administrator.
3. Verify that the WLS_WSM1 and WLS_WSM2 managed servers are listed.
4. Verify that the WSM-PM_Cluster cluster is listed.

5. Check that you can access Oracle Enterprise Manager at <http://SOAHOST1VHN1:7001/em>.
6. Log in to EM Console with the username and password you specified in [Section 4.6, "Creating boot.properties for the Administration Server on SOAHOST1."](#)

4.9 Applying the Java Required Files (JRF) Template to the WSM-PM_Cluster

After the domain is created with the Configuration Wizard, you must target a number of resources not included in the WebLogic server installation to the WSM-PM Cluster.

To target these resources:

1. Log in to Oracle Enterprise Manager Fusion Middleware Control with the username and password you specified in [Section 4.6, "Creating boot.properties for the Administration Server on SOAHOST1."](#)
2. On the navigation tree on the left, expand **Farm_<domain_name>**, **WebLogic Domain**, and then **<domain_name>**, and select **WSM-PM_Cluster**.
3. Click **Apply JRF Template** on the right.
4. Wait for the confirmation message to appear on the screen.

This message should confirm that the JRF Template has been successfully applied to the WSM-PM_Cluster cluster.

4.10 Disabling Host Name Verification for the Oracle WebLogic Administration Server and the WLS_WSM1 Managed Server

This step is required if you have not set up the appropriate certificates to authenticate the different nodes with the Administration Server (see [Chapter 7, "Setting Up Node Manager"](#)). If you have not configured the server certificates, you will receive errors when managing the different WebLogic Servers. To avoid these errors, disable host name verification while setting up and validating the topology, and enable it again once the EDG topology configuration is complete as described in [Chapter 7, "Setting Up Node Manager."](#)

Perform these steps to disable host name verification:

1. Log in to Oracle WebLogic Server Administration Console.
2. Click **Lock and Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page appears.
5. Select **AdminServer(admin)** in the Names column of the table. The Settings page for AdminServer(admin) appear.
6. Click the **SSL** tab.
7. Click **Advanced**.
8. Set Hostname Verification to **None**.
9. Click **Save**.
10. Repeat steps 4 to 8 for the WLS_WSM1 server.

11. Save and activate the changes.
12. The change requires restart of the Administration Server to be effective. To do this, complete these steps:
 - a. In the Summary of Servers screen, select the **Control** tab.
 - b. Select **AdminServer(admin)** in the table and then click **Shutdown**.
 - c. Start the Administration Server again from the command line as follows:

```
SOAHOST1> cd ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>/bin
SOAHOST1> ./startWebLogic.sh
```

4.11 Starting Node Manager on SOAHOST1

Perform these steps to start Node Manager on SOAHOST1:

1. Run the *setNMPProps.sh* script, which is located in the *ORACLE_COMMON_HOME/common/bin* directory, to set the *StartScriptEnabled* property to 'true' before starting Node Manager:

```
SOAHOST1> cd ORACLE_COMMON_HOME/common/bin
SOAHOST1> ./setNMPProps.sh
```

Note: You must use the *StartScriptEnabled* property to avoid class loading failures and other problems.

2. Start Node Manager:

```
SOAHOST1> cd WL_HOME/server/bin
SOAHOST1> ./startNodeManager.sh
```

4.12 Starting and Validating the WLS_WSM1 Managed Server

Perform these steps to start the WLS_WSM1 managed server and check that it is configured correctly:

1. Start the WLS_WSM1 managed server using the Oracle WebLogic Server Administration Console as follows:
 - a. Expand the **Environment** node in the Domain Structure window.
 - b. Choose **Servers**. The Summary of Servers page appears.
 - c. Click the **Control** tab.
 - d. Select **WLS_WSM1** and then click **Start**.
2. Verify that the server status is reported as "Running" in the Admin Console. If the server is shown as "Starting" or "Resuming," wait for the server status to change to "Started." If another status is reported (such as "Admin" or "Failed"), check the server output log files for errors. See [Section 11.5, "Troubleshooting"](#) for possible causes.
3. Access `http://SOAHOST1:7010/wsm-pm`.
4. Click **Validate Policy Manager**. A list of policies and assertion templates available in the data store appear.

Note: The configuration is incorrect if no policies or assertion templates appear.

4.13 Propagating the Domain Configuration to SOAHOST2 Using the unpack Utility

Perform these steps to propagate the domain configuration:

1. Run the following command on SOAHOST1 to copy the template file created previously.

```
SOAHOST1> cd MW_HOME/wc/common/bin
SOAHOST1> scp wcdomaintemplate.jar oracle@SOAHOST2:/MW_HOME/wc/common/bin
```

2. Run the `unpack` command on SOAHOST2 to unpack the propagated template.

Note: Run `unpack` from the `ORACLE_HOME/common/bin` directory, not from the `WL_HOME/common/bin` directory.

```
SOAHOST2> cd MW_HOME/wc/common/bin
SOAHOST2> ./unpack.sh -domain=ORACLE_BASE/admin/<domain_name>/mserver/<domain_name> -template=wcdomaintemplate.jar
```

Note: The `ORACLE_BASE/admin/<domain_name>/mserver` directory must exist before running `unpack`.

4.14 Disabling Host Name Verification for the WLS_WSM2 Managed Server

This step is required if you have not set up the appropriate certificates to authenticate the different nodes with the Administration Server (see [Chapter 7, "Setting Up Node Manager"](#)). If you have not configured the server certificates, you will receive errors when managing the different WebLogic Servers. To avoid these errors, disable host name verification while setting up and validating the topology, and enable it again once the EDG topology configuration is complete as described in [Chapter 7, "Setting Up Node Manager."](#)

Perform these steps to disable host name verification:

1. Log in to Oracle WebLogic Server Administration Console.
2. Click **Lock and Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page appears.
5. Select **WLS_WSM2** in the Names column of the table. The Settings page for AdminServer(admin) appear.
6. Click the **SSL** tab.
7. Click **Advanced**.
8. Set Hostname Verification to **None**.
9. Save and activate the changes.

4.15 Starting Node Manager on SOAHOST2

Perform these steps to start Node Manager on SOAHOST2:

1. Run the `setNMProps.sh` script, which is located in the `ORACLE_COMMON_HOME/common/bin` directory, to set the `StartScriptEnabled` property to 'true' before starting Node Manager:

```
SOAHOST2> cd ORACLE_COMMON_HOME/common/bin
SOAHOST2> ./setNMProps.sh
```

Note: You must use the `StartScriptEnabled` property to avoid class loading failures and other problems.

2. Start Node Manager:

```
SOAHOST2> cd WL_HOME/server/bin
SOAHOST2> ./startNodeManager.sh
```

4.16 Starting and Validating the WLS_WSM2 Managed Server

Perform these steps to start the WLS_WSM2 managed server and check that it is configured correctly:

1. Start the WLS_WSM2 managed server using the Administration Console.
2. Verify that the server status is reported as "Running" in the Admin Console. If the server is shown as "Starting" or "Resuming," wait for the server status to change to "Started." If another status is reported (such as "Admin" or "Failed"), check the server output log files for errors. See [Section 11.5, "Troubleshooting"](#) for possible causes.
3. Access `http://SOAHOST2:7010/wsm-pm`.
4. Click validate policy manager.

4.17 Configuring the Java Object Cache for Oracle WSM

The Java Object Cache (JOC) should be configured among all the servers running Oracle WSM. This local cache is provided to increase the performance of Oracle WSM.

The Java Object Cache can be configured using the `MW_HOME/oracle_common/bin/configure-joc.py` script. This is a Python script which can be used to configure JOC in the managed servers. The script runs in WLST online mode and expects the Administration Server to be up and running.

When configuring JOC ports for Oracle products, Oracle recommends using ports in the 9988 to 9998 range.

Note: After configuring the Java Object Cache using the `wlst` commands or `configure-joc.py` script, all affected managed servers should be restarted for the configurations to take effect.

Usage

1. Connect to the Administration Server using the command-line Oracle WebLogic Scripting Tool (WLST), for example:

```
MW_HOME/soa/common/bin/wlst.sh
```

```
$ connect()
```

Enter the Oracle WebLogic Administration user name and password when prompted.

2. After connecting to the Administration Server using `wlst`, start the script using the `execfile` command, for example:

```
wls:/mydomain/serverConfig>execfile('MW_HOME/oracle_
common/bin/configure-joc.py')
```

3. Configure JOC for all the managed servers for a given cluster.

Enter 'y' when the script prompts whether you want to specify a cluster name, and also specify the cluster name and discover port, when prompted. This discovers all the managed servers for the given cluster and configure the JOC. The discover port is common for the entire JOC configuration across the cluster. For example:

```
Do you want to specify a cluster name (y/n) <y>
Enter Cluster Name : WSM_Cluster
Enter Discover Port : 9991
```

Here is a walkthrough for using `configure-joc.py` for HA environments:

```
execfile('MW_HOME/oracle_common/bin/configure-joc.py')
.
Enter Hostnames (eg host1,host2) : SOAHOST1, SOAHOST2
.
Do you want to specify a cluster name (y/n) <y>
.
Enter Cluster Name : WSM_Cluster
.
Enter Discover Port : 9991
.
Enter Distribute Mode (true|false) <true> : true
.
Do you want to exclude any server(s) from JOC configuration (y/n) <n> n
```

The script can also be used to perform the following JOC configurations:

- Configure JOC for all specified managed servers.

Enter 'n' when the script prompts whether you want to specify a cluster name, and also specify the managed server and discover port, when prompted. For example:

```
Do you want to specify a cluster name (y/n) <y>n
Enter Managed Server and Discover Port (eg WLS_WSM1:9998, WLS_WSM1:9998) : WLS_
WSM1:9991,WLS_WSM2:9991
```

- Exclude JOC configuration for some managed servers.

The script allows you to specify the list of managed servers for which the JOC configuration "DistributeMode" will be set to 'false'. Enter 'y' when the script prompts whether you want to exclude any servers from JOC configuration, and enter the managed server names to be excluded, when prompted. For example:

```
Do you want to exclude any server(s) from JOC configuration (y/n) <n>y
Exclude Managed Server List (eg Server1,Server2) : WLS_WSM1,WLS_WSM3
```

- Disable the distribution mode for all managed servers.

The script allows you to disable the distribution to all the managed servers for a specified cluster. Specify 'false' when the script prompts for the distribution mode. By default, the distribution mode is set to 'true'.

Verify JOC configuration using the CacheWatcher utility. See *Oracle Fusion Middleware High Availability Guide*.

You can configure the Java Object Cache (JOC) using the **HA Power Tools** tab in the Oracle WebLogic Administration Console as described in the *Oracle Fusion Middleware High Availability Guide*.

4.18 Configuring Oracle HTTP Server for the Administration Server and the WLS_WSM-PMn Managed Servers

To enable Oracle HTTP Server to route to the Administration Server and the WSM-PM_Cluster, which contain the WLS_WSM-PMn managed servers, you must set the `WebLogicCluster` parameter to the list of nodes in the cluster:

1. On WEBHOST1 and WEBHOST2, add the following lines to the `ORACLE_BASE/admin/<instance_name>/config/OHS/<component_name>/mod_wl_ohs.conf` file:

```
# WSM-PM
<Location /wsm-pm>
    SetHandler weblogic-handler
    WebLogicCluster SOAHOST1:7010,SOAHOST2:7010
</Location>

# Admin Server and EM
<Location /console>
    SetHandler weblogic-handler
    WebLogicHost SOAHOST1VHN1
    WeblogicPort 7001
</Location>
<Location /consolehelp>
    SetHandler weblogic-handler
    WebLogicHost SOAHOST1VHN1
    WeblogicPort 7001
</Location>
<Location /em>
    SetHandler weblogic-handler
    WebLogicHost SOAHOST1VHN1
    WeblogicPort 7001
</Location>
```

2. Make sure the `httpd.conf` file located in the same directory as the `mod_wl_ohs` file contains the following lines:

```
NameVirtualHost *:7777
<VirtualHost *:7777>
    ServerName https://wc.mycompany.com:443
    ServerAdmin you@your.address
    RewriteEngine On
    RewriteOptions inherit
</VirtualHost>

NameVirtualHost *:7777
<VirtualHost *:7777>
    ServerName admin.mycompany.com:80
    ServerAdmin you@your.address
    RewriteEngine On
```

```
RewriteOptions inherit
</VirtualHost>
```

Note: Values such as 7777, `admin.mycompany:80`, and `you@youraddress` that are noted in this document serve as examples only. Enter values based on the actual environment.

3. Restart Oracle HTTP Server on both WEBHOST1 and WEBHOST2.

```
WEBHOST1> ORACLE_BASE/admin/<instance_name>/bin/opmnctl restartproc
ias-component=ohs1
```

```
WEBHOST2> ORACLE_BASE/admin/<instance_name>/bin/opmnctl restartproc
ias-component=ohs2
```

The servers specified in the `WebLogicCluster` parameter are only important at startup time for the plug-in. The list needs to provide at least one running cluster member for the plug-in to discover other members of the cluster. Note that the listed cluster member must be running when Oracle HTTP Server is started. Oracle WebLogic Server and the plug-in work together to update the server list automatically with new, failed, and recovered cluster members.

Some example scenarios:

- Example 1: If you have a two-node cluster and then add a third member, you do not need to update the configuration to add the third member. The third member will be discovered on the fly at runtime.
- Example 2: You have a three-node cluster but only two nodes are listed in the configuration. However, if both listed nodes are down when you start Oracle HTTP Server, then the plug-in would fail to route to the cluster. You must ensure that at least one of the listed nodes is running when you start Oracle HTTP Server.

If you list all members of the cluster, then you guarantee you can route to the cluster, assuming at least one member is running when Oracle HTTP Server is started.

For more information on configuring the WebLogic Server plug-in, see the Oracle Fusion Middleware Using Web Server Plug-Ins With Oracle WebLogic Server guide.

4.19 Registering Oracle HTTP Server With WebLogic Server

Oracle HTTP Server must forward requests to WebLogic Server. To do this, you must register Oracle HTTP Server with WebLogic Server using the following command:

```
WEBHOST1> cd ORACLE_BASE/admin/<instance_name>/bin
```

```
WEBHOST1> ./opmnctl registerinstance -adminHost SOAHOST1VHN1 -adminPort 7001
-adminUsername weblogic
```

You must also run this command from WEBHOST2 for OHS2.

4.20 Setting the Frontend URL for the Administration Console and Setting Redirection Preferences

When you access the Oracle WebLogic Server Administration Console using an LRB, changing the Administration Server's frontend URL is required so that the user's

browser is redirected to the appropriate LRB address. To change the Administration Server's frontend URL, complete these steps:

1. Log in to Oracle WebLogic Server Administration Console.
2. Click **Lock and Edit**.
3. Expand the Environment node in the Domain Structure window.
4. Click **Servers** to open the Summary of Servers page.
5. Select **Admin Server** in the Names column of the table. The Settings page for AdminServer(admin) appear.
6. Click the Protocols tab.
7. Click the HTTP tab.
8. Set the **Front End Hist** field to `soa.mycompany.com` (your LBR address).
9. Save and activate the changes.

Oracle also recommends disabling tracking on configuration changes in the Oracle WebLogic Server Administration Console so that the console does not trigger the reload of configuration pages when activation of changes occurs. To disable the reload, log in to the Oracle WebLogic Server Administration Console, click the **preferences** link in the banner, and then the **shared preferences** tab. Deselect the **follow configuration changes** checkbox.

4.21 Validating Access Through Oracle HTTP Server

Verify that the server status is reported as "Running" in the Admin Console. If the server is shown as "Starting" or "Resuming," wait for the server status to change to "Started." If another status is reported (such as "Admin" or "Failed"), check the server output log files for errors. See [Section 11.5, "Troubleshooting"](#) for possible causes.

Validate WSM_Cluster through both Oracle HTTP Server using the following URLs:

- `http://Webtier_node1:7777/wsm-pm`
- `http://Webtier_node2:7777/wsm-pm`
- `http://Webtier_node1:7777/console`
- `http://Webtier_node2:7777/console`
- `http://Webtier_node1:7777/em`
- `http://Webtier_node2:7777/em`
- `http://admin.mycompany.com/console`
- `http://admin.mycompany.com/em`

For information on configuring system access through the load balancer, see [Section 2.2.2, "Load Balancers."](#)

Note: After the registering Oracle HTTP Server as described in [Section 4.19, "Registering Oracle HTTP Server With WebLogic Server,"](#) the Oracle HTTP Server should appear as a manageable target in the Oracle Enterprise Manager Console. To verify this, log into the Enterprise Manager Console. The WebTier item in the navigation tree should show that Oracle HTTP Server has been registered.

4.22 Manually Failing Over the Administration Server to SOAHOST2

In case a node fails, you can fail over the Administration Server to another node. This section describes how to fail over the Administration Server from SOAHOST1 to SOAHOST2.

Assumptions:

- The Administration Server is configured to listen on SOAHOST1VHN1, and not on ANY address. See step 14 in [Section 4.4, "Running the Configuration Wizard on SOAHOST1 to Create a Domain"](#).
- These procedures assume that the two nodes use two individual domain directories, and that the directories reside in local storage or in shared storage in different volumes.
- The Administration Server is failed over from SOAHOST1 to SOAHOST2, and the two nodes have these IPs:
 - SOAHOST1: 100.200.140.165
 - SOAHOST2: 100.200.140.205
 - VIPHOST1: 100.200.140.206. This is the VIP where the Administration Server is running, assigned to ethX:Y, available in SOAHOST1 and SOAHOST2.
- The domain directory where the Administration Server is running in SOAHOST1 is on a shared storage and is mounted also from SOAHOST2.

The following procedure shows how to fail over the Administration Server to a different node (SOAHOST2), but the Administration Server will still use the same WebLogic Server machine (which is a logical machine, not a physical machine).

1. Stop the Administration Server.
2. Migrate IP to the second node.
 - a. Run the following command as root on SOAHOST1 (where X:Y is the current interface used by SOAHOST1VHN1):

```
SOAHOST1> /sbin/ifconfig ethX:Y down
```

- b. Run the following command on SOAHOST2:

```
SOAHOST2> /sbin/ifconfig <interface:index> <IP_Address> netmask <netmask>
```

For example:

```
/sbin/ifconfig eth0:1 10.0.0.1 netmask 255.255.255.0
```

Note: Ensure that the netmask and interface to be used to match the available network configuration in SOAHOST2.

3. Update routing tables through arping, for example:

```
SOAHOST2> /sbin/arping -b -A -c 3 -I eth0 10.0.0.1
```

4. Start the Administration Server on SOAHOST2.

```
SOAHOST2> cd ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>/bin
```

```
SOAHOST2> ./startWebLogic.sh
```

5. Test that you can access the Administration Server on SOAHOST2 as follows:

- a. Ensure that you can access the Oracle WebLogic Server Administration Console at `http://SOAHOST1VHN1:7001/console`.
- b. Check that you can access and verify the status of components in the Oracle Enterprise Manager at `http://SOAHOST1VHN1:7001/em`.

Note: The Administration Server does not use Node Manager for failover. After a manual failover, the machine name that appears in the **Current Machine** field in the Administration Console for the server is SOAHOST1, and not the failover machine, SOAHOST2. Since Node Manager does not monitor the Administration Server, the machine name that appears in the **Current Machine** field, is not relevant and you can ignore it.

4.23 Validating Access to SOAHOST2 Through Oracle HTTP Server

Perform the same steps as in [Section 4.21, "Validating Access Through Oracle HTTP Server"](#). This is to check that you can access the Administration Server when it is running on SOAHOST2.

4.24 Failing the Administration Server Back to SOAHOST1

This step checks that you can fail back the Administration Server, that is, stop it on SOAHOST2 and run it on SOAHOST1. To do this, migrate SOAHOST1VHN1 back to SOAHOST1 node as follows:

1. Run the following command on SOAHOST2.

```
SOAHOST2> /sbin/ifconfig ethZ:N down
```

2. Run the following command on SOAHOST1:

```
SOAHOST1> /sbin/ifconfig ethX:Y 100.200.140.206 netmask 255.255.255.0
```

Note: Ensure that the netmask and interface to be used match the available network configuration in SOAHOST1

3. Update routing tables through arping. Run the following command from SOAHOST1.

```
SOAHOST1> /sbin/arping -b -A -c 3 -I ethZ 100.200.140.206
```

4. Start the Administration Server again on SOAHOST1.

```
SOAHOST1> cd ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>/bin
SOAHOST1> ./startWebLogic.sh
```

5. Test that you can access the Oracle WebLogic Server Administration Console at `http://SOAHOST1VHN1:7001/console`.
6. Check that you can access and verify the status of components in the Oracle Enterprise Manager at `http://SOAHOST1VHN1:7001/em`.

4.25 Backing Up the Installation

Perform a backup to save your domain configuration (make sure that you stop the server first). The configuration files all exist under the `ORACLE_BASE/admin/<domain_name>` directory.

```
SOAHOST1> tar -cvpf edgdomainback.tar ORACLE_BASE/admin/<domain_name>
```

Extending the Domain for SOA Components

This chapter describes how to use the Configuration Wizard to extend the domain to include SOA components. You created in the domain in [Chapter 4, "Creating a Domain."](#)

Important: Oracle strongly recommends that you read the release notes for any additional installation and deployment considerations prior to starting the setup process.

Note: Follow the steps in this chapter only if you want to run SOA components on SOAHOST1 and SOAHOST2. If you do not want to run SOA components in your WebCenter topology, you can skip this chapter.

This chapter contains the following sections:

- [Section 5.1, "Installing Oracle Fusion Middleware for SOA Home"](#)
- [Section 5.2, "Enabling SOAHOST1VHN2 and SOAHOST2VHN1"](#)
- [Section 5.3, "Extending the Domain for SOA Components"](#)
- [Section 5.4, "Restarting the Administration Server"](#)
- [Section 5.5, "Configuring Oracle Coherence for Deploying Composites"](#)
- [Section 5.6, "Running the soa-createUDD.py Script"](#)
- [Section 5.7, "Disabling Host Name Verification for the WLS_SOAn Managed Server"](#)
- [Section 5.8, "Restarting the Node Manager on SOAHOST1"](#)
- [Section 5.9, "Propagating the Domain Changes to the Managed Server Domain Directory"](#)
- [Section 5.10, "Starting the WLS_SOA1 Managed Server on SOAHOST1"](#)
- [Section 5.11, "Validating the WLS_SOA1 and WLS_WSM1 Managed Servers"](#)
- [Section 5.12, "Propagating the Domain Configuration to SOAHOST2 Using the unpack Utility"](#)
- [Section 5.13, "Extracting the XEngine Files in SOAHOST2"](#)
- [Section 5.14, "Restarting Node Manager on SOAHOST2"](#)

- [Section 5.15, "Starting and Validating the WLS_SOA2 Managed Server"](#)
- [Section 5.16, "Configuring Oracle HTTP Server for the WLS_SOA Managed Servers"](#)
- [Section 5.17, "Validating Access Through Oracle HTTP Server"](#)
- [Section 5.18, "Setting the Frontend HTTP Host and Port"](#)
- [Section 5.19, "Configuring a Shared JMS Persistence Store"](#)
- [Section 5.20, "Configuring a Default Persistence Store for Transaction Recovery"](#)
- [Section 5.21, "Enabling High Availability for Oracle File and FTP Adapters"](#)
- [Section 5.22, "Scaling the Oracle Database Adapter"](#)
- [Section 5.23, "Backing Up the Installation"](#)

5.1 Installing Oracle Fusion Middleware for SOA Home

You must install Oracle Fusion Middleware for SOA on both SOAHOST1 and SOAHOST2. These nodes will run managed servers configured with SOA components.

1. Start the installer for Oracle Fusion Middleware for SOA.

```
SOAHOST1> runInstaller
```

When the installer prompts you for a JRE/JDK location enter the Oracle SDK location created in the Oracle WebLogic Server installation, for example, **MW_HOME/jrockit_160_14_R27.6.4-18**.

2. In the Welcome screen, click **Next**.
3. In the Prerequisite Check screen, verify that the checks complete successfully, and click **Next**.
4. Specify the installation location. Select the previously installed Middleware Home from the drop-down list. For the Oracle Home directory, enter the directory name (soa).

Click **Next**.

5. In the Installation Summary screen, click **Install**.
6. In the Installation Complete screen, click **Finish**.

5.2 Enabling SOAHOST1VHN2 and SOAHOST2VHN1

The SOA domain uses virtual hostnames as the listen addresses for the SOA managed servers. You must enable A VIP mapping each of these hostnames on the two SOA Machines, (VIP2 on SOAHOST1 and VIP3 on SOAHOST2), and must be correctly resolve the virtual hostnames in the network system used by the topology (either by DNS Server, hosts resolution).

To enable the VIP, follow the steps described in [Section 4.3, "Enabling VIP1 in SOAHOST1"](#). These VIPs and VHNs are required to enable server migration for the SOA Servers. Server migration must be configured for the SOA System for high availability purposes. Refer to [Chapter 8, "Configuring External Services"](#) for more details on configuring server migration for the SOA servers.

5.3 Extending the Domain for SOA Components

In this step, you extend the domain created in [Chapter 4, "Creating a Domain"](#) to contain SOA components. In this section we are assuming that the SOA deployment uses the same database service (wcedg.mycompany.com) as the WebCenter deployment. However, a deployment may choose to use a different database service specifically for SOA such as *soaedg.mycompany.com*.

Note: You must back up the current domain before extending the domain. You may use the backup to recover in case any errors were made in the domain extension. See *Oracle Fusion Middleware Administrator's Guide*.

Note: Oracle SOA uses Quartz to maintain its jobs and schedules in the database. The system clocks for the SOA WebLogic cluster must be synchronized to enable Quartz jobs to run correctly across the cluster.

1. Change directory to the location of the Configuration Wizard. This is within the SOA home directory. (It is recommended that all database instances should be up.)

```
SOAHOS1> cd ORACLE_HOME/common/bin
```

2. Start the Configuration Wizard.

```
SOAHOST1> ./config.sh
```

3. In the Welcome screen, select **Extend an existing WebLogic domain**, and click **Next**.

4. In the WebLogic Domain Directory screen, select the WebLogic domain directory (ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>), and click **Next**.

5. In the Select Extension Source screen, do the following:

- Select **Extend my domain automatically to support the following added products**.
- Select the following products:
 - **Business Rules**
 - **User Messaging Service**
 - **User Messaging Service Drivers for SMS, IM, Voice**
 - **User Messaging Service Worklist Drivers**
 - **Oracle SOA Suite 11.1.1.0**

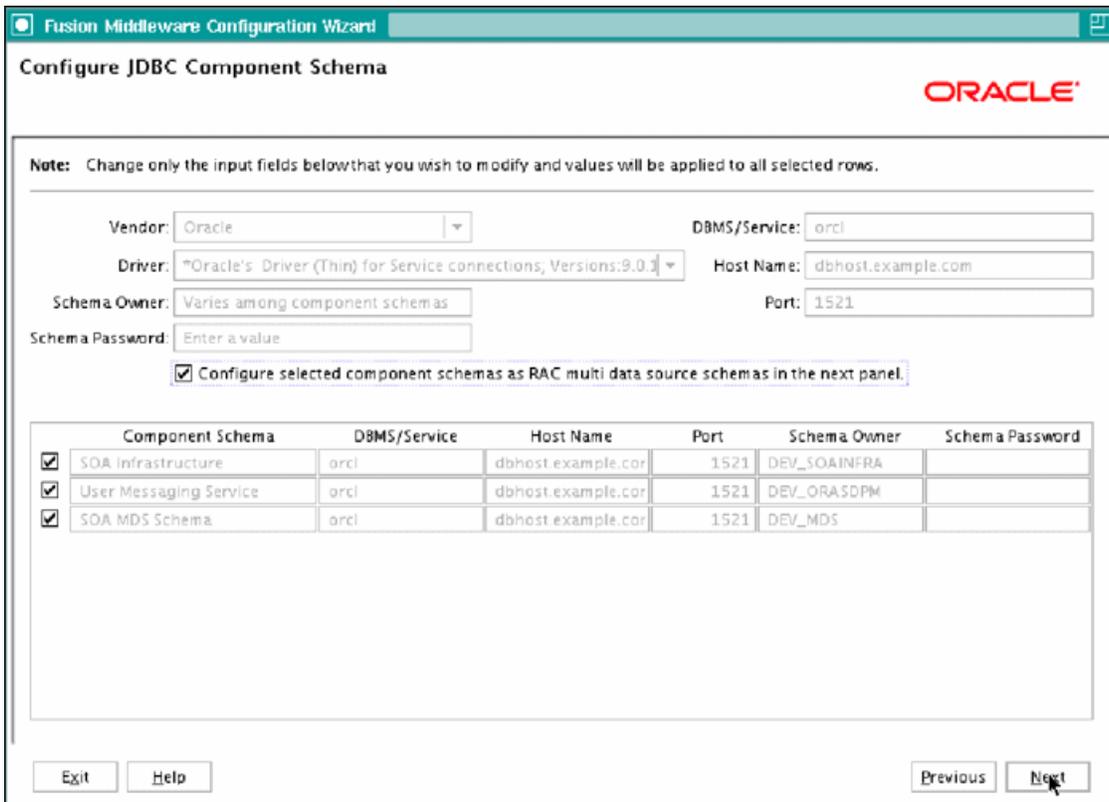
The following products should already be selected, and grayed out. They were selected when you created in domain in [Section 4.4, "Running the Configuration Wizard on SOAHOST1 to Create a Domain."](#)

- Basic WebLogic Server Domain
- JRF

Click **Next**.

6. If you get a "Conflict Detected" message that Oracle JRF is already defined in the domain, select the **Keep Existing Component** option and click **OK**.
7. In the Configure JDBC Component Schema screen (Figure 5–1), do the following:
 - a. Select the **SOA Infrastructure**, **User Messaging Service**, and **SOA MDS Schema** rows in the table.
 - b. Select **Configure selected component schemas as RAC multi data source schemas in the next panel**.
 - c. Click **Next**.

Figure 5–1 Configure JDBC Component Schema Screen

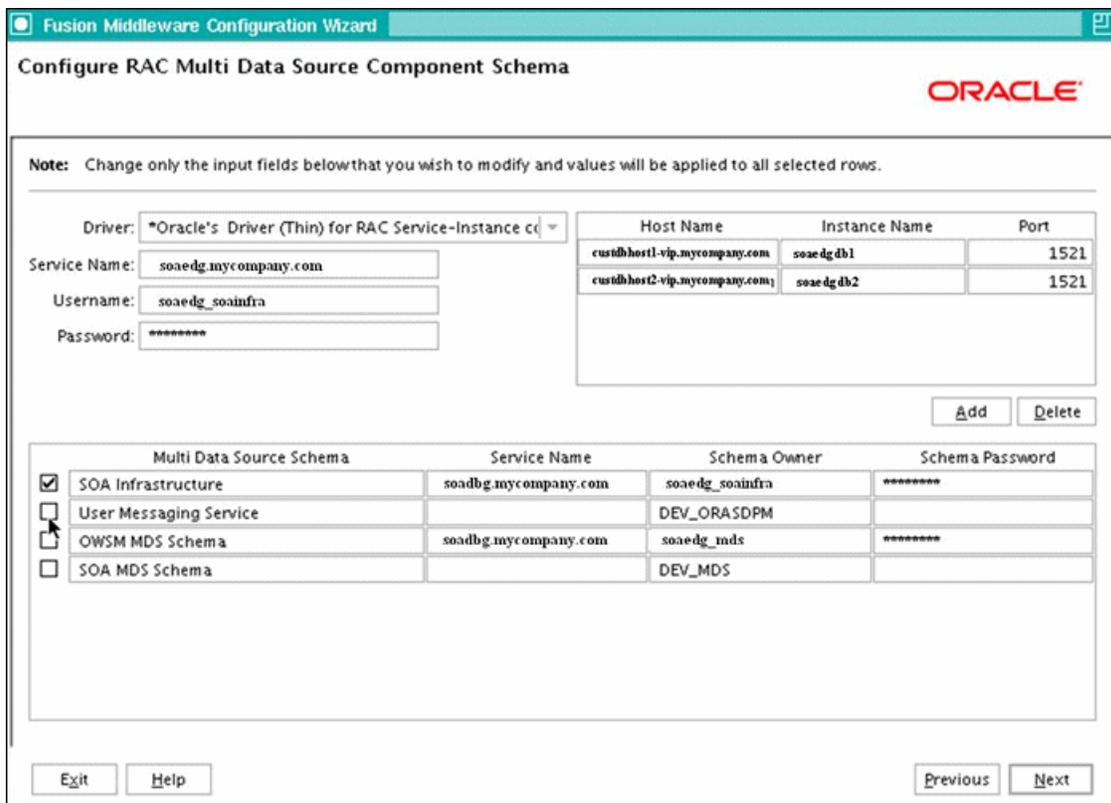


Screenshot of the Configure JDBC Component Schema screen for SOA.

8. In the Configure RAC Multi Data Source Component Schema screen (Figure 5–2), do the following:
 - a. Select **SOA Infrastructure**.
 - b. Enter values for the following fields, specifying the connect information for the RAC database that was seeded with RCU.
 - **Driver:** Select **Oracle driver (Thin) for RAC Service-Instance connections, Versions:10, 11**.
 - **Service Name:** Enter the service name of the database; for example, `wcedg.mycompany.com`.

- **Username:** Enter the complete user name (including prefix) for the schemas. The user names shown in [Figure 5-2](#) assume that soedg was used as prefix for schema creation from RCU.
 - **Password:** Enter the password to use to access the schemas.
- c. Click **Add** and enter the details for the first RAC instance.
 - d. Repeat for each RAC instance.
 - e. Deselect **SOA Infrastructure**.
 - f. Select **User Messaging Service**.
 - g. Repeat steps b, c, and d for the User Messaging Schema.
 - h. Deselect **User Messaging Service**.
 - i. Select **SOA MDS Schema**.
 - j. Repeat steps b, c, and d for the SOA MDS Schema.
 - k. Leave the OWSM MDS Schema information as it is.
 - l. Click **Next**

Figure 5-2 Configure RAC Multi Data Source Component Schema Screen



Screenshot of the Configure RAC Multi Data Source Component Schema screen for SOA.

9. In the Test JDBC Data Sources screen, the connections should be tested automatically. The Status column displays the results. Ensure that all connections

were successful. If not, click **Previous** to return to the previous screen and correct your entries.

Click **Next** when all the connections are successful.

10. In the Select Optional Configuration screen, select **Managed Servers, Cluster and Machines**, and then **Deployment and services**. Click **Next**.

11. In the Configure Managed Servers screen, add the required managed servers.

A server called `soa_server1` is created automatically. Rename this to `WLS_SOA1` and give it the attributes listed in [Table 5-1](#). Then, add a new server called `WLS_SOA2`. The `WLS_WSM1` and `WLS_WSM2` managed servers should already be present because they are part of the domain that you are extending. In the end, the list of managed servers should match that in [Table 5-1](#).

Table 5-1 Managed Servers

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled
WLS_SOA1	WLS_SOA1	8001	n/a	No
WLS_SOA2	WLS_SOA2	8001	n/a	No
WLS_WSM1	SOAHOST1	7010	n/a	No
WLS_WSM2	SOAHOST2	7010	n/a	No

Click **Next**.

12. In the Configure Clusters screen, add the following clusters:

Table 5-2 Clusters

Name	Cluster Messaging Mode	Multicast Address	Multicast Port	Cluster Address
SOA_Cluster	unicast	n/a	n/a	Leave it empty.
WSM-PM_Cluster	unicast	n/a	n/a	Leave it empty.

Click **Next**.

13. In the Assign Servers to Clusters screen, assign servers to clusters as follows:

- **SOA_Cluster:**
 - WLS_SOA1
 - WLS_SOA2
- **WSM-PM_Cluster:**
 - WLS_WSM1
 - WLS_WSM2

Click **Next**.

14. In the Configure Machines screen, do the following:

- Delete the **LocalMachine** that appears by default.
- Click the **Unix Machine** tab. The following entries appear (listed in [Table 5-3](#)):

Table 5-3 Machines

Name	Node Manager Listen Address
SOAHOST1	SOAHOST1
SOAHOST2	SOAHOST2

Leave all other fields to their default values.

Click **Next**.

15. In the Assign Servers to Machines screen, assign servers to machines as follows:

- **SOAHOST1:**
 - AdminServer
 - WLS_SOA1
 - WLS_WSM1
- **SOAHOST2:**
 - WLS_SOA2
 - WLS_WSM2

Click **Next**.

16. In the Target Deployments to Clusters or Servers screen, ensure the following targets:

- **usermessagingserver** and **usermessagingdriver-email** should be targeted only to **SOA_Cluster**. (The **usermessaging-xmpp**, **usermessaging-smpp**, and **usermessaging-voicexml** applications are optional.)
- The **oracle.rules.***, **oracle.sdp.***, and **oracle.soa.*** deployments should be targeted only to **SOA_Cluster**.
- The **wsm-smp** application should be targeted only to **WSM-PM_Cluster**.
- The **oracle.rules.***, **oracle.sdp.***, and **oracle.soa.*** libraries should be targeted only to **SOA_Cluster**.
- The **oracle.wsm.seedpolicies** library should be targeted only to **WSM-PM_Cluster**.

Click **Next**.

17. In the Target Services to Clusters or Servers screen, ensure the following targets:

- Target **JOC Startup Class** and **JOC Shutdown Class** only to **WSM-PM_Cluster**.
- Target **OWSM Startup Class** only to **WSM-PM_Cluster**.
- Target **mds-owsm**, **mds-owsm-rac0**, and **mds-owsm-rac1** to both **WSM-PM_Cluster** and **AdminServer**.

Click **Next**.

18. In the Configuration Summary screen click **Extend**.

Note: Click **OK** to dismiss the warning dialog about the domain configuration ports conflicting with the host ports. This warning appears because of the existing WSM-PM installation.

19. In the Extending Domain screen, click **Done**.

You must restart the Administration Server for this configuration to take effect.

5.4 Restarting the Administration Server

Restart the Administration Server so that the changes to the domain are picked up.

1. Stop the Administration Server.

```
SOAHOST1> cd ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>/bin
SOAHOST1> ./stopWebLogic.sh
```

2. Start the Administration Server:

```
SOAHOST1> ./startWebLogic.sh
```

5.5 Configuring Oracle Coherence for Deploying Composites

Although deploying composites uses multicast communication by default, Oracle recommends using unicast communication instead in SOA enterprise deployments. Use unicast if you disable multicast communication for security reasons.

Note: An incorrect configuration of the Oracle Coherence framework that is used for deployment may prevent the SOA system from starting. The deployment framework must be properly customized for the network environment on which the SOA system runs. Oracle recommends the following configuration described in this section.

Enabling Communication for Deployment Using Unicast Communication

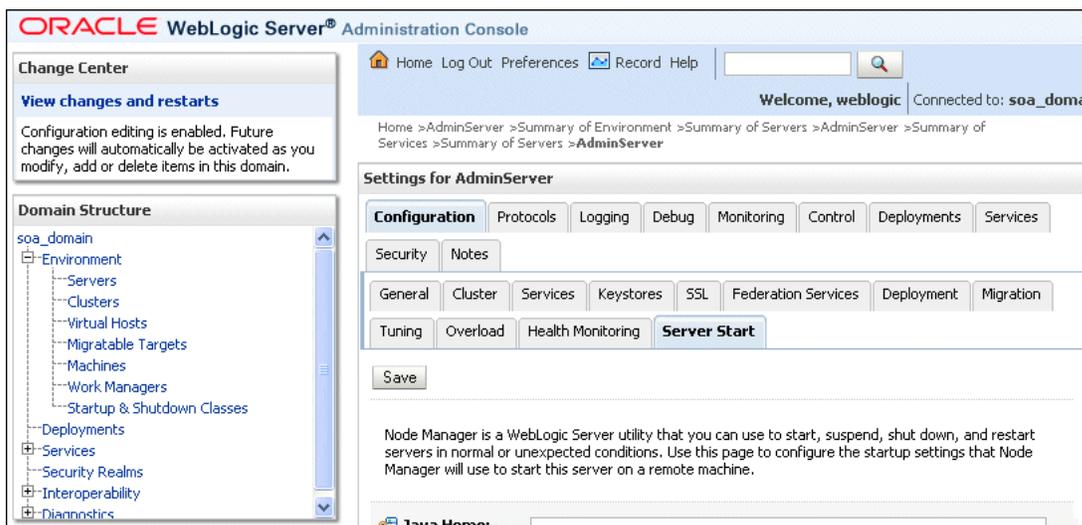
Multicast communication enables Oracle Fusion Middleware SOA to discover all of the members of a cluster to which it deploys composites dynamically. However, unicast communication does not enable nodes to discover other cluster members in this way. Consequently, you must specify the nodes that belong to the cluster. You do not need to specify all of the nodes of a cluster, however. You need only specify enough nodes so that a new node added to the cluster can discover one of the existing nodes. As a result, when a new node has joined the cluster, it is able to discover all of the other nodes in the cluster. Additionally, in configurations such as SOA enterprise deployments where multiple IPs are available in the same box, you must configure Oracle Coherence to use a specific host name to create the Oracle Coherence cluster.

Tip: To guarantee high availability during deployments of SOA composites, specify enough nodes so that at least one of them is running at any given time.

Specify the nodes using the `tangosol.coherence.wka<n>` system property, where `<n>` is a number between 1 and 9. You can specify up to 9 nodes. Start the numbering at 1. This numbering must be sequential and must not contain gaps. In addition, specify the host name used by Oracle Coherence to create a cluster through the `tangosol.coherence.localhost` system property. This local host name should be the virtual host name used by the SOA server as the listener addresses (SOAHOST1VHN2 and SOAHOST2VHN1). Set this property by adding the `-Dtangosol.coherence.localhost` parameters to the Arguments field of the Oracle WebLogic Server Administration Console's Server Start tab (Figure 5-3).

Note: SOAHOST1VHN2 is the virtual host name that maps to the virtual IP where WLS_SOA1 listening (in SOAHOST1). SOAHOST2VHN1 is the virtual host name that maps to the virtual IP where WLS_SOA2 is listening (in SOAHOST2).

Figure 5–3 *Setting the Host Name Using the Start Server Tab of Oracle WebLogic Server Administration Console*



This image shows the Server Start tab of the Oracle WebLogic Server Administration Console. The surrounding text describes how to access this tab and how to use it.

Specifying the host name

To add the host name used by Oracle Coherence, complete these steps:

1. Log into the Oracle WebLogic Server Administration Console.
2. In the Domain Structure window, expand the **Environment** node.
3. Click **Servers**. The Summary of Servers page appears.
4. Click the name of the server (**WLS_SOA1** or **WLS_SOA2**, which are represented as hyperlinks) in Name column of the table. The settings page for the selected server appears.
5. Click **Lock and Edit**.
6. Click the Configuration tab, and then the Server Start tab.
7. Enter the following for WLS_SOA1 and WLS_SOA2 into the Arguments field.

For WLS_SOA1, enter the following:

```
-Dtangosol.coherence.wka1=SOAHOST1VHN2 -Dtangosol.coherence.wka2=SOAHOST2VHN1
-Dtangosol.coherence.localhost=SOAHOST1VHN2
```

For WLS_SOA2, enter the following:

```
-Dtangosol.coherence.wka1=SOAHOST2VHN1 -Dtangosol.coherence.wka2=SOAHOST1VHN2
-Dtangosol.coherence.localhost=SOAHOST2VHN1
```

8. Click **Save** and **Activate Changes**.

Note: You must ensure that these variables are passed to the managed server correctly. (They should be reflected in the server's output log.) Failure of the Oracle Coherence framework can prevent the soa-infra application from starting.

Note: The multicast and unicast addresses are different from the ones used by the WebLogic Server cluster for cluster communication. SOA guarantees that composites are deployed to members of a single WebLogic Server cluster even though the communication protocol for the two entities (the WebLogic Server cluster and the groups to which composites are deployed) are different.

Note: The coherence cluster used for deployment uses port 8088 by default. This port can be changed by specifying the `-Dtangosol.coherence.wkaX.port` startup parameter.

5.6 Running the soa-createUDD.py Script

Before you can start the WLS_SOA1 managed server on SOAHOST1, you must run the `soa-createUDD.py` script. You run the script using WLST offline:

1. Run the `setDomainEnv.sh` script to set up the environment:

```
SOAHOST1>. ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>/bin/
setDomainEnv.sh
```

2. Run the `soa-createUDD.py` script, which is located in the `ORACLE_HOME/bin` directory:

```
SOAHOST1> MW_HOME/jrockit_160_14_R27.6.4-18/bin/java weblogic.WLST ORACLE_
HOME/bin/soa-createUDD.py --domain_home ORACLE_BASE/admin/<domain_
name>/aserver/<domain_name> --soacluster SOA_Cluster
```

3. Restart the Administration Server. See the steps in [Section 5.4, "Restarting the Administration Server."](#)

4. Verify that the following modules are listed in the Oracle WebLogic Server Administration Console:

- **SOAJMSModuleUDDs**

Verify this module by first expanding the **Services** node in the Domain Structure window and then expanding the **Messaging** node in the Oracle WebLogic Server Administration Console. Select **JMS Modules**. Select **SOAJMSModuleUDDs** (represented as a hyperlink) in the Names column of the table. Open the **Subdeployment** tab, and then click **SOAJMSSubDM**.

Verify that the module is targeted to `SOAJMSServer_auto_1` and `SOAJMSServer_auto_2`.

- **UMSJMSSystemResource**

Verify this module by first expanding the **Services** node in the Domain Structure window and then expanding the **Messaging** node in the Oracle

WebLogic Server Administration Console. Select **JMS Modules**. Select **UMSJMSSystemResource** (represented as a hyperlink) in the Names column of the table. Open the **Subdeployment** tab, and then click **UMSJMSSubDMSOA**.

Verify that the module is targeted to UMSJMSServer_auto_1 and UMSJMSServer_auto_2.

You also create UDDs using the **HA Power Tools** tab in the Oracle WebLogic Administration Console as described in the *Oracle Fusion Middleware High Availability Guide*

5.7 Disabling Host Name Verification for the WLS_SOAn Managed Server

This step is required if you have not set up the appropriate certificates to authenticate the different nodes with the Administration Server (see [Chapter 7, "Setting Up Node Manager"](#)). If you have not configured the server certificates, you will receive errors when managing the different WebLogic Servers. To avoid these errors, disable host name verification while setting up and validating the topology, and enable it again once the EDG topology configuration is complete as described in [Chapter 7, "Setting Up Node Manager."](#)

To disable host name verification, complete these steps:

1. Log in to Oracle WebLogic Server Administration Console.
2. Click **Lock and Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page appears.
5. Select **WLS_SOA1** (represented as a hyperlink) from the Names column of the table. The Settings page appears.
6. Select the **SSL** tab.
7. Expand the **Advanced** section of the page.
8. Set Hostname Verification to **None**.
9. Click **Save**.
10. Repeat these steps for the WLS_SOA2 managed server.
11. Save and activate the changes.

5.8 Restarting the Node Manager on SOAHOST1

To restart the Node Manager on SOAHOST1, complete these steps:

1. Stop Node Manager.


```
SOAHOST1> cd WL_HOME/server/bin
SOAHOST1> ./stopNodeManager.sh
```
2. Start Node Manager:


```
SOAHOST1> ./startNodeManager.sh
```

5.9 Propagating the Domain Changes to the Managed Server Domain Directory

To propagate the start scripts and classpath configuration from the Administration Server's domain directory to the managed server domain directory:

1. Create a copy of the managed server domain directory and the managed server applications directory.
2. Run the `pack` command on SOAHOST1 to create a template pack using the following commands:

```
SOAHOST1> cd ORACLE_HOME/common/bin
```

```
SOAHOST1> ./pack.sh -managed=true -domain=ORACLE_BASE/admin/<domain_name>/aserver/<domain_name> -template=soadomaintemplateExtSOA.jar -template_name=soa_domain_templateExtSOA
```

3. Run the `unpack` command on SOAHOST1 to unpack the propagated template to the domain directory of the managed server using the following command:

```
SOAHOST1> ./unpack.sh -domain=ORACLE_BASE/admin/<domain_name>/msserver/<domain_name>/ -overwrite_domain=true -template=soadomaintemplateExtSOA.jar -app_dir=ORACLE_BASE/admin/<domain_name>/msserver/apps
```

5.10 Starting the WLS_SOA1 Managed Server on SOAHOST1

To start the WLS_SOA1 managed server on SOAHOST1, complete these steps:

1. Access the Administration Console at <http://SOAHOST1VHN1:7001/console>.
2. Click **Servers**.
3. Open the **Control** tab.
4. Select **WLS_SOA1**.
5. Click **Start**.

Note: SOAHOST1VHN1 is the virtual host name that maps to the virtual IP where the Administration Server is listening (in SOAHOST1).

5.11 Validating the WLS_SOA1 and WLS_WSM1 Managed Servers

To validate the WLS_SOA1 and WLS_WSM1 managed servers, complete these steps:

1. Verify that the server status is reported as "Running" in the Admin Console. If the server is shown as "Starting" or "Resuming," wait for the server status to change to "Started." If another status is reported (such as "Admin" or "Failed"), check the server output log files for errors. See [Section 11.5, "Troubleshooting"](#) for possible causes.
2. Access <http://SOAHOST1VHN2:8001/soa-infra> to verify status of WLS_SOA1.
3. Access <http://SOAHOST1:7010/wsm-pm> to verify status of WLS_WSM1.
4. Access <http://SOAHOST1VHN2:8001/b2b> to verify status of B2B.
5. Access <http://SOAHOST1VHN2:8001/integration/worklistapp> to verify status of the worklist application.

Note: These URLs vary if you use VIPs for WLS_SOA*n*.

5.12 Propagating the Domain Configuration to SOAHOST2 Using the unpack Utility

To propagate the domain configuration, complete these steps:

1. Run the following command on SOAHOST1 to copy the template file created in the previous step to SOAHOST2.

```
SOAHOST1> cd ORACLE_HOME/common/bin
```

```
SOAHOST1> scp soadomaintemplateExtSOA.jar oracle@node2:ORACLE_HOME/common/bin
```

2. Run the unpack command on SOAHOST2 to unpack the propagated template.

```
SOAHOST2> cd ORACLE_HOME/common/bin
```

```
SOAHOST2> ./unpack.sh -domain=ORACLE_BASE/admin/<domain_name>/mserver/<domain_name>/ -template=soadomaintemplateExtSOA.jar -overwrite_domain=true -app_dir=ORACLE_BASE/admin/<domain_dir>/mserver/apps
```

5.13 Extracting the XEngine Files in SOAHOST2

To enable B2B's XEngine in SOAHOST2, you need to extract the content of the XEngine tar manually:

```
SOAHOST2>cd ORACLE_HOME/soa/thirdparty/edifecs
```

```
SOAHOST2>tar -xzvf XEngine.tar.gz
```

5.14 Restarting Node Manager on SOAHOST2

Perform the steps in [Section 5.8, "Restarting the Node Manager on SOAHOST1"](#) on SOAHOST2.

5.15 Starting and Validating the WLS_SOA2 Managed Server

Perform these steps to start the WLS_SOA2 managed server and check that it is configured correctly:

1. Start the WLS_SOA2 managed server using the Administration Console.
2. Verify that the server status is reported as "Running" in the Admin Console. If the server is shown as "Starting" or "Resuming," wait for the server status to change to "Started." If another status is reported (such as "Admin" or "Failed"), check the server output log files for errors. See [Section 11.5, "Troubleshooting"](#) for possible causes.
3. Access <http://SOAHOST2VHN1:8001/soa-infra>.
4. Access <http://SOAHOST2VHN1:8001/b2b> to verify status of B2B.
5. Access <http://SOAHOST2VHN1:8001/integration/worklistapp> to verify status of the worklist application.

5.16 Configuring Oracle HTTP Server for the WLS_SOAn Managed Servers

To enable Oracle HTTP Server to route to the SOA_Cluster, which contains the WLS_SOAn managed servers, you must set the `WebLogicCluster` parameter to the list of nodes in the cluster.

1. On `WEBHOST1` and `WEBHOST2`, add the following lines to the `ORACLE_BASE/admin/<instance_name>/config/OHS/<component_name>/mod_wl_ohs.conf` file:

```
# SOA soa-infra app
<Location /soa-infra>
    SetHandler weblogic-handler
    WebLogicCluster SOAHOST1VHN2:8001,SOAHOST2VHN1:8001
</Location>

# Worklist
<Location /integration/>
    SetHandler weblogic-handler
    WebLogicCluster SOAHOST1VHN2:8001,SOAHOST2VHN1:8001
</Location>

# B2B
<Location /b2b>
    SetHandler weblogic-handler
    WebLogicCluster SOAHOST1VHN2:8001,SOAHOST2VHN1:8001
</Location>

# UMS prefs
<Location /sdpMessaging/userprefs-ui >
    SetHandler weblogic-handler
    WebLogicCluster SOAHOST1VHN2:8001,SOAHOST2VHN1:8001
</Location>

# Default to-do taskflow
<Location /DefaultToDoTaskFlow/>
    SetHandler weblogic-handler
    WebLogicCluster SOAHOST1VHN2:8001,SOAHOST2VHN1:8001
</Location>

# Workflow
<Location /workflow>
    SetHandler weblogic-handler
    WebLogicCluster SOAHOST1VHN2:8001,SOAHOST2VHN1:8001
</Location>

#Required if attachments are added for workflow tasks
<Location /ADFAttachmentHelper>
    SetHandler weblogic-handler
    WebLogicCluster SOAHOST1VHN2:8001,SOAHOST2VHN1:8001
</Location>

# SOA composer application
<Location /soa/composer>
    SetHandler weblogic-handler
    WebLogicCluster SOAHOST1VHN2:8001,SOAHOST2VHN1:8001
</Location>
```

Note: The entry for `/workflow` is optional. It is for workflow tasks associated with ADF task forms. The `/workflow` URL itself can be a different value, depending on the form.

2. Restart Oracle HTTP Server on WEBHOST1 and WEBHOST2:

```
WEBHOST1> ORACLE_BASE/admin/<instance_name>/bin/opmnctl restartproc
ias-component=ohs1
WEBHOST2> ORACLE_BASE/admin/<instance_name>/bin/opmnctl restartproc
ias-component=ohs2
```

The servers specified in the `WebLogicCluster` parameter are only important at startup time for the plug-in. The list needs to provide at least one running cluster member for the plug-in to discover other members of the cluster. Note that the listed cluster member must be running when Oracle HTTP Server is started. Oracle WebLogic Server and the plug-in work together to update the server list automatically with new, failed, and recovered cluster members.

Sample scenarios:

- Example 1: If you have a two-node cluster and then add a third member, you do not need to update the configuration to add the third member. The third member will be discovered on the fly at runtime.
- Example 2: You have a three-node cluster but only two nodes are listed in the configuration. However, if both listed nodes are down when you start Oracle HTTP Server, then the plug-in would fail to route to the cluster. You must ensure that at least one of the listed nodes is running when you start Oracle HTTP Server.

If you list all members of the cluster, then you guarantee you can route to the cluster, assuming at least one member is running when Oracle HTTP Server is started.

For more information on configuring the WebLogic Server plug-in, see the Oracle Fusion Middleware Using Web Server Plug-Ins With Oracle WebLogic Server guide.

5.17 Validating Access Through Oracle HTTP Server

Verify that the server status is reported as "Running" in the Admin Console. If the server is shown as "Starting" or "Resuming," wait for the server status to change to "Started." If another status is reported (such as "Admin" or "Failed"), check the server output log files for errors. See [Section 11.5, "Troubleshooting"](#) for possible causes.

Verify that you can access these URLs, where 'webhostN' specifies the name of each Oracle HTTP Server host (for example, WEBHOST1, WEBHOST2):

- `http://webhostN:7777/soa-infra`
- `http://webhostN:7777/integration/worklistapp`
- `http://webhostN:7777/b2b`
- `http://webhostN:7777/sdpMessaging/userprefs-ui`

Validate SOA_Cluster through both Oracle HTTP Server instances.

Refer to load balancer configuration to access the system through the load balancer.

5.18 Setting the Frontend HTTP Host and Port

You must set the frontend HTTP host and port for the Oracle WebLogic Server cluster:

1. In the WebLogic Server Administration Console, in the Change Center section, click **Lock & Edit**.
2. In the left pane, choose **Environment** in the Domain Structure window and then choose **Clusters**. The Summary of Clusters page appears.
3. Select the WLS_SOA cluster.
4. Select **HTTP**.
5. Set the values for the following:
 - **Frontend Host:** wc.mycompany.com
 - **Frontend HTTPS Port:** 443
6. Click **Save**.
7. To activate the changes, click **Activate Changes** in the Change Center section of the Administration Console.

Note: If you do not set the frontend HTTP host and port, you get the following message when trying to retrieve a document definition XSD from Oracle B2B:

```
An error occurred while loading the document definitions.  
java.lang.IllegalArgumentException: Cluster address must be set  
when clustering is enabled.
```

Callback URL

The SOA system calculates the callback URL as follows:

- If a request to SOA originates from an external or internal service, then SOA uses the callback URL specified by the client.
- If a request to an external or internal asynchronous service originates from SOA, the callback URL is determined using the following method, in decreasing order of preference:
 1. Use `callbackServerURL` specified as a binding property for the specific reference. (You can set this when modeling the composite or at runtime using the MBeans). This allows different service calls to have different callback URLs. That is, a callback URL from an external service can be set to be different than one to an internal service. In the context of the EDG architecture, typically this will be `soa.mycompany.com (443/https)` for external services and `soainternal.mycompany.com (7777/http)` for internal services. At runtime, this property is set using the System MBean Browser, through the corresponding binding mbean. To add a specific URL, add a `callbackServerURL` property to its Properties attribute, then invoke the save operation.
 2. Use the callback URL as specified in `soa-infra-config.xml`. In this case, only one address can be specified. When a mix of both external and internal services can be invoked, this should be set to `soa.mycompany.com (443/https)` in the EDG architecture. When only internal services are to be invoked, this can be set to `soainternal.mycompany.com (7777/http)`.

3. Use the callback URL as the frontend host specified in WLS for the SOA_Cluster. In this case, too, only one address can be specified and the recommendation is same as the one for *soa-infra-config.xml*.
4. Use the local host name as provided by WLS MBean APIs. This is not recommended in HA environments such as EDG.

5.19 Configuring a Shared JMS Persistence Store

Configure the location for all of the persistence stores as a directory that is visible from both nodes. For more information see [Section 4.1, "Installing Oracle Fusion Middleware Home."](#) You must then change all of the persistent stores to use this shared base directory as follows:

1. Log into the Oracle WebLogic Server Administration Console.
2. In the Domain Structure window, expand the **Services** node and then click the **Persistence Stores** node. The Summary of Persistence Stores page appears.
3. Select the persistence store (represented as a hyperlink) from the Name column of the table. The Settings page for the persistence store appear.
4. In the Configuration tab, enter the location on a persistent storage solution (such as NAS or SAN) that is available to other servers in the cluster in the Directory field. Specifying this location enables pending JMS messages to be sent. The location should follow the following directory structure:

```
ORACLE_BASE/admin/<domain_name>/<soa_cluster_name>/jms
```

Note: Both WLS_SOA1 and WLS_SOA2 must be able to access this directory. This directory must also exist before you restart the server.

5. Click **Save** and activate changes.
6. Restart the servers to make the change in the persistent stores effective.

5.20 Configuring a Default Persistence Store for Transaction Recovery

Each server has a transaction log that stores information about committed transactions that are coordinated by the server that may not have been completed. The WebLogic Server uses this transaction log for recovery from system crashes or network failures. To leverage the migration capability of the Transaction Recovery Service for the servers within a cluster, store the transaction log in a location accessible to a server and its backup servers.

Note: Preferably, this location should be a dual-ported SCSI disk or on a Storage Area Network (SAN).

To set the location for the default persistence store, complete these steps:

1. Log into the Oracle WebLogic Server Administration Console.
2. In the Domain Structure window, expand the **Environment** node and then click the **Servers** node. The Summary of Servers page appears.

3. Click the name of the server (represented as a hyperlink) in Name column of the table. The settings page for the selected server appears and defaults to the Configuration tab.
4. Click the **Services** tab.
5. In the Default Store section of the page, enter the path to the folder where the default persistent stores will store its data files. The directory structure of the path is as follows:

```
ORACLE_BASE/admin/<domain_name>/<soa_cluster_name>/tlogs
```

6. Click **Save**.

Note: To enable migration of the Transaction Recovery Service, specify a location on a persistent storage solution that is available to other servers in the cluster. Both WLS_SOA1 and WLS_SOA2 must be able to access this directory. This directory must also exist before you restart the server.

5.21 Enabling High Availability for Oracle File and FTP Adapters

The Oracle File and FTP Adapters enable a BPEL process or an Oracle Mediator to read and write files on local file systems and on remote file systems through FTP (File Transfer Protocol). These adapters support high availability for an active-active topology with Oracle BPEL Process Manager and Oracle Mediator service engines for both inbound and outbound operations. To make Oracle File and FTP Adapters highly available for outbound operations, use the database mutex locking operation as described in "High Availability in Outbound Operations" in *Oracle Fusion Middleware User's Guide for Technology Adapters*. The database mutex locking operation enables these adapters to ensure that multiple references do not overwrite one another if they write to the same directory.

Note: The File Adapter picks up a file from the inbound directory, processes it, and then outputs a file to the output directory. Because the File Adapter is non-transactional, files can be processed twice. As a result, it is possible to get duplicate files when there is failover in the RAC backend or in the SOA managed servers.

5.21.1 Using the Database Mutex Locking Operation

Use the following procedure to make an outbound Oracle File or FTP Adapter service highly available using database table as a coordinator:

Note: You must increase global transaction timeouts if you use database as a coordinator.

1. **Create Database Tables**

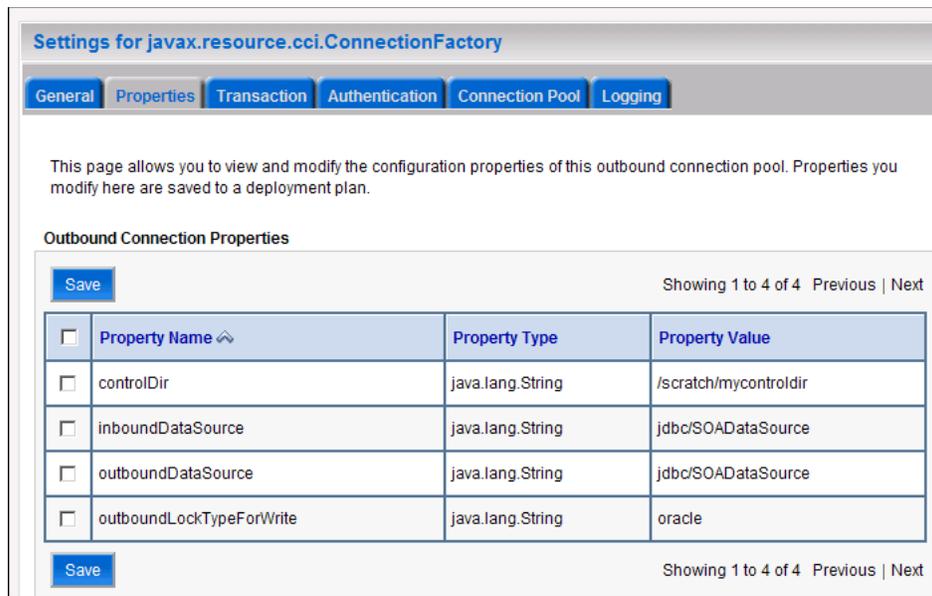
You are not required to perform this step since the database schemas are pre-created as a part of soainfra.

2. **Modify Deployment Descriptor for Oracle File Adapter**

Modify Oracle File Adapter deployment descriptor for the connection-instance corresponding to `eis/HAFileAdapter` from the Oracle WebLogic Server console:

- a. Log into your Oracle WebLogic Server console. To access the console navigate to `http://servername:portnumber/console`.
- b. Click **Deployments** in the left pane for Domain Structure.
- c. Click **FileAdapter** under Summary of Deployments on the right pane.
- d. Click the **Configuration** tab.
- e. Click the **Outbound Connection Pools** tab, and expand **javax.resource.cci.ConnectionFactory** to see the configured connection factories, as shown in [Figure 5-4](#).
- f. Click **eis/HAFileAdapter**. The Outbound Connection Properties for the connection factory corresponding to high availability is displayed.
- g. The connection factory properties appear as shown in [Figure 5-4](#).

Figure 5-4 Oracle WebLogic Server Console - Settings for javax.resource.cci.Connectionfactory Page



This figure shows the Oracle WebLogic Server Console - Settings for `javax.resource.cci.ConnectionFactory` page with the **Properties** tab selected. The connection property name, type, and value columns provide the relevant connection property information.

Click on **Lock and Edit**. After this, the property value column becomes editable (you can click on any of the rows under "Property Value" and modify its value).

The new parameters in connection factory for Oracle File and FTP Adapters are as follows:

`controlDir`: Set it to the directory structure where you want the control files to be stored. You must set it to a shared location if multiple WebLogic Server instances run in a cluster. Structure the directory for shared storage as follows:

```
ORACLE_BASE/admin/<domain_name>/<cluster_name>/fadapter
```

`inboundDataSource`: Set the value to `jdbc/SOADDataSource`. This is the data source, where the schemas corresponding to high availability are pre-created. The pre-created schemas can be found under `ORACLE_HOME/rcu/integration/soainfra/sql/adapter/createschema_adapter_oracle.sql`. If you want to create the schemas elsewhere, use this script. You must set the `inboundDataSource` property accordingly if you choose a different schema.

`outboundDataSource`: Set the value to `jdbc/SOADDataSource`. This is the data source where the schemas corresponding to high availability are pre-created. The pre-created schemas can be found under `ORACLE_HOME/rcu/integration/soainfra/sql/adapter/createschema_adapter_oracle.sql`. If you want to create the schemas elsewhere, use this script. You must set the `outboundDataSource` property if you choose to do so.

`outboundLockTypeForWrite`: Set the value to `oracle` if you are using Oracle Database. By default the Oracle File and FTP Adapters use an in-memory mutex to lock outbound write operations. You must choose from the following values for synchronizing write operations:

`memory`: The Oracle File and FTP Adapters use an in-memory mutex to synchronize access to the file system.

`oracle`: The adapter uses Oracle Database sequence.

`db`: The adapter uses a pre-created database table (`FILEADAPTER_MUTEX`) as the locking mechanism. You must use this option only if you are using a schema other than the Oracle Database schema.

`user-defined`: The adapter uses a user-defined mutex. To configure the user-defined mutex, you must implement the mutex interface: `"oracle.tip.adapter.file.Mutex"` and then configure a new binding-property with the name `"oracle.tip.adapter.file.mutex"` and value as the fully qualified class name for the mutex for the outbound reference.

- h. Click **Save** after you update the properties. The Save Deployment Plan page appears.
- i. Enter a shared storage location for the deployment plan. The directory structure is as follows:

```
ORACLE_BASE/admin/<domain_name>/<cluster_name>/dp/Plan.xml
```

- j. Click **Save and Activate**.
- k. Configure BPEL Process or Mediator Scenario to use the connection factory as shown in the following example:

```
<adapter-config name="FlatStructureOut" adapter="File Adapter"
xmlns="http://platform.integration.oracle/blocks/adapter/fw/metadata">
  <connection-factory location="eis/HAFFileAdapter" adapterRef="" />
  <endpoint-interaction portType="Write_ptt" operation="Write">
<interaction-spec
className="oracle.tip.adapter.file.outbound.FileInteractionSpec">
  <property../>
```

```

        <property../>
    </interaction-spec>
</endpoint-interaction>
</adapter-config>

```

Note: The location attribute is set to `eis/HADFileAdapter` for the connection factory.

5.22 Scaling the Oracle Database Adapter

The best practice for multiple Oracle Database Adapter process instances that are deployed to multiple Oracle BPEL Process Manager or Oracle Mediator nodes is to use `LogicalDeletePollingStrategy` or `DeletePollingStrategy` with a unique `MarkReservedValue` on each polling node and setting `MaxTransactionSize`. [Example 5-1](#) illustrates how to use `MarkReservedValue` with `DeletePollingStrategy` in a `.jca` file.

Example 5-1 Snippet of `.jca` Showing the Usage of `MarkReservedValue` with `DeletePollingStrategy`

```

<adapter-config name="poll" adapter="Database Adapter"
xmlns="http://platform.integration.oracle/blocks/adapter/fw/metadata">
<connection-factory location="eis/DB/SOADemo" UIConnectionName="SOADemo"
adapterRef="" />
<endpoint-activation portType="poll_ptt" operation="receive">
    <activation-spec className="oracle.tip.adapter.db.DBActivationSpec">
        <property name="DescriptorName" value="poll.PerfMasterIn" />
        <property name="QueryName" value="pollSelect" />
        <property name="MappingsMetaDataURL" value="poll-or-mappings.xml" />
        <property name="PollingStrategy" value="DeletePollingStrategy" />
        <property name="MarkReadColumn" value="ATTRIBUTE2" />
        <property name="MarkReservedValue"
value=MyPrefix_${weblogic.Name-1}-${IP}" />
        <property name="MarkUnreadValue" value="UNREAD" />
        <property name="PollingInterval" value="5" />
        <property name="MaxRaiseSize" value="5" />
        <property name="MaxTransactionSize" value="10" />
        <property name="SequencingColumn" value="PK" />
        <property name="ReturnSingleResultSet" value="false" />
    </activation-spec>
</endpoint-activation>
</adapter-config>

```

For more information, see "Scalability" and "Polling Strategies" in *Oracle Fusion Middleware User's Guide for Technology Adapters*.

5.23 Backing Up the Installation

After you have verified that the extended domain is working, back up the installation. This is a quick backup for the express purpose of immediate restore in case of problems in the further steps. The backup destination is the local disk. This backup can be discarded once the enterprise deployment setup is complete. At this point, the regular deployment-specific backup and recovery process can be initiated. The *Oracle Fusion Middleware Administrator's Guide* provides further details. For information on describing the Oracle HTTP Server data that must be backed up and restored, refer to the "Backup and Recovery Recommendations for Oracle HTTP Server" section in this guide. For information on how to recover components, see "Recovery of Components"

and "Recovery After Loss of Component" sections in the guide. For recommendations specific to recovering from the loss of a host, see the "Recovering Oracle HTTP Server to a Different Host" in the guide. Also refer to the *Oracle Database Backup and Recovery Guide* for information on database backup.

To back up the installation at this point, complete these steps:

1. Back up the web tier:

- a. Shut down the instance using `opmnctl`.

```
ORACLE_BASE/admin/<instance_name>/bin/opmnctl stopall
```

- b. Back up the Middleware Home on the web tier using the following command (as root):

```
tar -cvpf BACKUP_LOCATION/web.tar $MW_HOME
```

- c. Back up the Instance Home on the web tier using the following command (as root):

```
tar -cvpf BACKUP_LOCATION/web_instance.tar $ORACLE_INSTANCE
```

- d. Start the instance using `opmnctl`:

```
ORACLE_BASE/admin/<instance_name>/bin/opmnctl startall
```

2. Back up the database. This is a full database backup (either hot or cold) using Oracle Recovery Manager (recommended) or OS tools such as `tar` for cold backups if possible.

3. Back up the Administration Server domain directory to save your domain configuration. The configuration files all exist under the `ORACLE_BASE/admin/<domain_name>` directory.

```
SOAHOST1> tar -cvpf edgdomainback.tar ORACLE_BASE/admin/<domain_name>
```

Note: `ORACLE_HOME` should be backed up if any changes are made to the XEngine configuration that are part of your B2B setup. These files are located under `ORACLE_HOME/soa/thirdparty/edifecs/XEngine`. To back up `ORACLE_HOME`, execute the following command:

```
SOAHOST1> tar -cvpf fmwhomeback.tar MW_HOME
```

Extending the Domain for WebCenter Components

This chapter describes how to use the Configuration wizard to extend the domain that you created in [Chapter 4, "Creating a Domain."](#) It contains the following sections:

- [Section 6.1, "Installing Oracle Fusion Middleware Home"](#)
- [Section 6.2, "Extending the Domain for WebCenter Components"](#)
- [Section 6.3, "Restarting the Administration Server"](#)
- [Section 6.4, "Disabling Host Name Verification for the WebCenter Managed Servers"](#)
- [Section 6.5, "Starting Node Manager on SOAHOST1"](#)
- [Section 6.6, "Propagating the Domain Changes to the Managed Server Domain Directory"](#)
- [Section 6.7, "Propagating the Domain Configuration to SOAHOST2, WCHOST1, and WCHOST2 Using the unpack Utility"](#)
- [Section 6.8, "Starting the Node Manager on WCHOST1 and WCHOST2"](#)
- [Section 6.9, "Starting the WLS_Spaces1, WLS_Portlet1, and WLS_Services1 Managed Servers on WCHOST1"](#)
- [Section 6.10, "Validating the WLS_Spaces1, WLS_Portlet1, and WLS_Services1 Managed Servers"](#)
- [Section 6.11, "Starting the WLS_Spaces2, WLS_Portlet2, and WLS_Services2 Managed Servers on WCHOST2"](#)
- [Section 6.12, "Validating the WLS_Spaces2, WLS_Portlet2, and WLS_Services2 Managed Servers"](#)
- [Section 6.13, "Setting Up the Java Object Cache"](#)
- [Section 6.14, "Converting Discussions Forum from Multicast to Unicast"](#)
- [Section 6.15, "Configuring Clustering for Discussion Server"](#)
- [Section 6.16, "Configuring WS-Security for Discussion Server"](#)
- [Section 6.17, "Configuring Wiki Server"](#)
- [Section 6.18, "Configuring Oracle HTTP Server for the WLS_Spacesn, WLS_Portletn, and WLS_Servicesn Managed Servers on WCHOST2"](#)
- [Section 6.19, "Validating Access Through Oracle HTTP Server"](#)
- [Section 6.20, "Validating Access Through the Load Balancer"](#)

- [Section 6.21, "Backing Up the Installation"](#)

6.1 Installing Oracle Fusion Middleware Home

You must install Oracle Fusion Middleware on WCHOST1 and WCHOST2. These nodes will run managed servers configured with WebCenter components.

Follow the steps in [Section 4.1, "Installing Oracle Fusion Middleware Home"](#). You must install both Oracle WebLogic Server and WebCenter.

6.2 Extending the Domain for WebCenter Components

In this step, you extend the domain created in [Chapter 4, "Creating a Domain"](#) to contain WebCenter components.

Note: You must back up the current domain before extending the domain. You may use the backup to recover in case any errors were made in the domain extension. See *Oracle Fusion Middleware Administrator's Guide*.

1. Change directory to the location of the Configuration wizard. This is within the WebCenter home directory.

```
SOAHOST1> cd MW_HOME/wc/common/bin
```

2. Start the Configuration Wizard.

```
SOAHOST1> ./config.sh
```

Note: If you run the Configuration Wizard from the same shell and environment that you used to create the domain, you must deselect the `CONFIG_JVM_ARGS=-DTemplateCatalog.enable.selectable.all=true` variable. Otherwise, the Configuration Wizard displays all of the available templates, which are not needed for extending the domain for WebCenter components.

3. In the Welcome screen, select **Extend an existing WebLogic domain**, and click **Next**.
4. In the WebLogic Domain Directory screen, Select the WebLogic domain directory (`ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>`), and click **Next**.
5. In the Select Extension Source screen, do the following:
 - Select **Extend my domain automatically to support the following added products**.
 - Select the following products:
 - **WebCenter Portlet Producers**
 - **WebCenter Discussion Server**
 - **Wiki and Blog Server**
 - **WebCenter Spaces**

The following products should already be selected, and grayed out. They were selected when you created in domain in [Section 4.4, "Running the Configuration Wizard on SOAHOST1 to Create a Domain."](#)

- Basic WebLogic Server Domain
- JRF
- WSM-PM

Click **Next**.

6. If you get a "Conflict Detected" message that Oracle JRF is already defined in the domain, select the **Keep Existing Component** option and click **OK**.
7. In the Configure JDBC Data Sources screen, do the following:
 - a. Ensure that the following data sources appear on the screen. The user names shown in [Table 6-1](#) assume that wcedg was used as prefix for schema creation from RCU.

Table 6-1 Values for Data Sources

Data Source	User Name
mds-SpacesDS	wcedg_mds
WebCenterDS	wcedg_webcenter
PortletDS	wcedg_portlet
DiscussionDS	wcedg_discussions
OWCWikiDS	wcedg_wiki

- b. Select the check box next to all the component schemas.
 - c. Select **Configure all datasources as RAC multi-datasources in the next panel**.
 - d. Click **Next**.
8. Configure RAC Multi Data Sources screen
 - a. Enter values for the following fields, specifying the connect information for the RAC database that was seeded with RCU.
 - **Driver:** Select **Oracle driver (Thin) for RAC Service-Instance connections, Versions:10, 11**.
 - **Service Name:** Enter the service name of the database, for example, `wcedg.mycompany.com`.
 - **Username prefix:** Enter the user name for each of the schemas by selecting each schema individually. The user names shown in [Table 6-1](#) assume that wcedg was used as prefix for schema creation from RCU.
 - **Password and Confirm Password:** enter the password to use to access the schemas.
 - b. Click **Add** and enter the details for the first RAC instance.
 - c. Repeat for each RAC instance.
 - d. Click **Next**.
9. In the Test JDBC Data Sources screen, the connections should be tested automatically. The **Status** column displays the results. Ensure that all connections

were successful. If not, click **Previous** to return to the previous screen and correct your entries.

Click **Next** when all the connections are successful.

10. In the Advanced Configuration Screen, select the following:

- Managed Servers, Clusters and Machines
- Deployments and Services

Click **Next**.

11. In the Configure Managed Servers screen, add the following managed servers:

Table 6–2 Managed Servers

Name	Server	Listen Port	SSL Listen Port	SSL Enabled
WLS_Spaces1	WCHOST1	9000	n/a	No
WLS_Spaces2	WCHOST2	9000	n/a	No
WLS_Portlet1	WCHOST1	9001	n/a	No
WLS_Portlet2	WCHOST2	9001	n/a	No
WLS_Services1	WCHOST1	9002	n/a	No
WLS_Services2	WCHOST2	9002	n/a	No

Note: Providing the listen address is mandatory if the cluster mode is 'unicast'.

Click **Next**.

12. The Configure Clusters screen should already include the WSM_PM Cluster in the list. Add the following three new clusters:

Table 6–3 Clusters

Name	Cluster Messaging Mode	Multicast Address	Multicast Port	Cluster Address
Spaces_Cluster	unicast	n/a	n/a	Leave it empty.
Portlet_Cluster	unicast	n/a	n/a	Leave it empty.
Services_Cluster	unicast	n/a	n/a	Leave it empty.

Click **Next**.

13. In the Assign Servers to Clusters screen, assign servers to clusters as follows:

- **Spaces_Cluster:**
 - WLS_Spaces1
 - WLS_Spaces2
- **Portlet_Cluster:**
 - WLS_Portlet1
 - WLS_Portlet2
- **Services_Cluster:**

- WLS_Services1
- WLS_Services2

Click **Next**.

14. In the Configure Machines screen, click the Unix Machine tab. Make sure the following four entries exist:

Table 6-4 Machines

Name	Node Manager Listen Address
SOAHOST1	SOAHOST1VHN2 Note that SOAHOST1 should already be configured when you ran the Configuration wizard in Section 4.4, "Running the Configuration Wizard on SOAHOST1 to Create a Domain."
SOAHOST2	SOAHOST2VHN1 Note that SOAHOST2 should already be configured when you ran the Configuration wizard in Section 4.4, "Running the Configuration Wizard on SOAHOST1 to Create a Domain."
WCHOST1	WCHOST1
WCHOST2	WCHOST2

Leave all other fields to their default values.

Click **Next**.

15. In the Assign Servers to Machines screen, assign servers to machines as follows:

- **SOAHOST1:**
 - AdminServer
 - WLS_WSM1
- **SOAHOST2:**
 - WLS_WSM2
- **WCHOST1:**
 - WLS_Spaces1
 - WLS_Portlet1
 - WLS_Services1
- **WCHOST2:**
 - WLS_Spaces2
 - WLS_Portlet2
 - WLS_Services2

Click **Next**.

16. In the Target Deployment to Clusters or Servers screen, click **Next**.
17. In the Review WebLogic Domain screen, click **Next**.
18. In the Extend WebLogic Domain screen, do not change the values that appear on the screen (since you are extending a domain). Click **Extend**.
19. In the Extending Domain screen, click **Done**.

6.3 Restarting the Administration Server

Restart the Administration Server so that the changes to the domain are picked up.

1. Stop the Administration Server.

```
SOAHOST1> ./stopWebLogic.sh
```

2. Start the Administration Server:

```
SOAHOST1> ./startWebLogic.sh
```

6.4 Disabling Host Name Verification for the WebCenter Managed Servers

Before you can start and verify the managed servers, you must disable host name verification. You can re-enable it after you set up SSL communication between the Administration Server and the Node Manager.

To disable host name verification, complete these steps:

1. Expand the **Environment** node in the Oracle WebLogic Server Administration Console.
2. Select **Servers**. The Summary of Servers page appears.
3. Select **WLS_Spaces1** (represented as a hyperlink) from the Names column of the table. The Settings page appears.
4. Select the **SSL** tab.
5. Expand the **Advanced** section of the page.
6. Set Hostname Verification to **None**.
7. Repeat these steps for the WLS_Spaces2, WLS_Portlet1, WLS_Portlet2, WLS_Services1, and WLS_Services2 managed servers.

6.5 Starting Node Manager on SOAHOST1

Make sure that Node Manager was started on SOAHOST1. If this is not the case, start Node Manager:

```
SOAHOST1> cd WL_HOME/server/bin
SOAHOST1> ./startNodeManager.sh
```

6.6 Propagating the Domain Changes to the Managed Server Domain Directory

As described in [Section 2-2, "Directory Structure,"](#) we need to separate the Administration Server domain directory from the managed server directories. In this step, we propagate the changes from one to the other. To propagate the start scripts and classpath configuration from the Administration Server's domain directory to the managed server domain directory, complete these steps:

1. Create a copy of the managed server domain directory and the managed server applications directory.
2. Move these directories using the following command (both commands):

```
mv ORACLE_BASE/admin/<domain_name>/msserver/apps ORACLE_BASE/admin/<domain_
name>/msserver/appsbackup
```

```
mv ORACLE_BASE/admin/<domain_name>/mserver/<domain_name> ORACLE_
BASE/admin/<domain_name>/mserver/>domain_name>backup
```

3. Run the `pack` command on SOAHOST1 to create a template pack using the following commands:

```
SOAHOST1> cd MW_HOME/wc/common/bin
```

```
SOAHOST1> ./pack.sh -managed=true -domain=ORACLE_BASE/admin/<domain_
name>/aserver/<domain_name> -template=wcdomaintemplateExtWC.jar -template_
name=wc_domain_templateExtWC
```

4. Run the `unpack` command on SOAHOST1 to unpack the propagated template to the domain directory of the managed server using the following command:

```
SOAHOST1> ./unpack.sh -domain=ORACLE_BASE/admin/<domain_name>/mserver/<domain_
name>/ -template=wcdomaintemplateExtWC.jar -overwrite_domain=true -app_
dir=ORACLE_BASE/admin/<domain_name>/mserver/apps
```

6.7 Propagating the Domain Configuration to SOAHOST2, WCHOST1, and WCHOST2 Using the `unpack` Utility

To propagate the domain configuration, complete these steps:

1. Run the following commands on SOAHOST1 to copy the template file created earlier to SOAHOST2, WCHOST1, and WCHOST:

```
SOAHOST1> scp wcdomaintemplate.jar
oracle@SOAHOST2:MW_HOME/wc/common/bin
```

```
SOAHOST1> scp wcdomaintemplate.jar
oracle@WCHOST1:MW_HOME/wc/common/bin
```

```
SOAHOST1> scp wcdomaintemplate.jar
oracle@WCHOST2:MW_HOME/wc/common/bin
```

2. Run the `unpack` command on SOAHOST2, WCHOST1, and WCHOST2 to unpack the propagated template.

```
SOAHOST2> cd MW_HOME/wc/common/bin
```

```
SOAHOST2> ./unpack.sh
-domain=ORACLE_BASE/admin/<domain_name>/mserver/<domain_name>/
-app_dir=ORACLE_BASE/admin/<domain_dir>/mserver/apps
```

3. Repeat the above steps for WCHOST1 and WCHOST2.

6.8 Starting the Node Manager on WCHOST1 and WCHOST2

1. Set the `JAVA_OPTIONS` environment variable to define the `StartScriptEnabled` property before starting Node Manager. On WCHOST1 and WCHOST2, run the following command:

```
WCHOST1> export JAVA_OPTIONS=-DStartScriptEnabled=true
WCHOST2> export JAVA_OPTIONS=-DStartScriptEnabled=true
```

2. Start Node Manager:

```
WCHOST1> cd WL_HOME/server/bin
WCHOST1> ./startNodeManager.sh
```

```
WCHOST2> cd WL_HOME/server/bin
WCHOST2> ./startNodeManager.sh
```

After you have started Node Manager for the first time, you can edit the `nodemanager.properties` file to set the `StartScriptEnabled` property. The `nodemanager.properties` file does not exist until Node Manager is started for the first time.

To set the `StartScriptEnabled` property in the `nodemanager.properties` file:

- Add the following line to the `WL_HOME/common/nodemanager/nodemanager.properties` file:

```
StartScriptEnabled=true
```

When this property is set in the `nodemanager.properties` file, you no longer need to define it in the `JAVA_OPTIONS` environment variable.

6.9 Starting the WLS_Spaces1, WLS_Portlet1, and WLS_Services1 Managed Servers on WCHOST1

Follow these steps to start the WLS_Spaces1, WLS_Portlet1, and WLS_Services1 managed servers:

1. Access the Administration Console at `http://SOAHOST1VHN1:7001/console`.
2. Click **Servers**.
3. Open the **Control** tab.
4. Select **WLS_Spaces1**, **WLS_Portlet1**, and **WLS_Services1**.
5. Click **Start**.

Note: SOAHOST1VHN1 is the virtual hostname that maps to the virtual IP where the Administration Server is listening (in SOAHOST1).

6.10 Validating the WLS_Spaces1, WLS_Portlet1, and WLS_Services1 Managed Servers

1. Check that the managed servers are accessible by testing the following URLs:
 - `http://WCHOST1:9000/webcenter`
 - `http://WCHOST1:9001/portalTools`
 - `http://WCHOST1:9002/owc_wiki`
 - `http://WCHOST1:9001/wsrp-info`
 - `http://WCHOST1:9001/richtextportlet`
 - `http://WCHOST1:9002/owc_discussions`
2. Check that all deployments are active. In the Administration Console, select **Deployments**. If any failed, check the log files for any errors. The log files can be found at `ORACLE_BASE/admin/<domain_name>/mserver/<domain_home>/servers/[server_name]/logs`.

6.11 Starting the WLS_Spaces2, WLS_Portlet2, and WLS_Services2 Managed Servers on WCHOST2

Follow these steps to start the WLS_Spaces2, WLS_Portlet2, and WLS_Services2 managed servers:

1. Access the Administration Console at `http://SOAHOST1VHN1:7001/console`.
2. Click **Servers**.
3. Open the **Control** tab.
4. Select **WLS_Spaces2, WLS_Portlet2, and WLS_Services2**.
5. Click **Start**.

6.12 Validating the WLS_Spaces2, WLS_Portlet2, and WLS_Services2 Managed Servers

1. Check that the managed servers are accessible by testing the following URLs:
 - `http://WCHOST2:9000/webcenter`
 - `http://WCHOST2:9001/portalTools`
 - `http://WCHOST2:9001/wsrp-tools`
 - `http://WCHOST2:9002/owc_wiki`
 - `http://WCHOST2:9002/owc_discussions`
2. Check that all deployments are active. In the Administration Console, select **Deployments**. If any failed, check the log files for any errors. The log files can be found at `ORACLE_BASE/admin/<domain_name>/mserver/<domain_home>/servers/[server_name]/logs`.

6.13 Setting Up the Java Object Cache

The Java Object Cache (JOC) should be configured among all the servers running WebCenter Spaces. This local cache is provided to increase the performance of Oracle WebCenter Spaces.

The Java Object Cache can be configured using the `MW_HOME/oracle_common/bin/configure-joc.py` script. This is a Python script which can be used to configure JOC in the managed servers. The script runs in WLST online mode and expects Administration Server to be up and running.

Note: After configuring the Java Object Cache using the `wlst` commands or `configure-joc.py` script, all affected managed servers should be restarted for the configurations to take effect.

Usage

1. Connect to the Administration Server using the command-line Oracle WebLogic Scripting Tool (WLST), for example:

```
MW_HOME/wc/common/bin/wlst.sh
$ connect()
```

Enter the Oracle WebLogic Administration user name and password when prompted.

2. After connecting to the Administration Server using `wlst`, start the script using the `execfile` command, for example:

```
wls:/mydomain/serverConfig>execfile('FMW_HOME/oracle_
common/bin/configure-joc.py')
```

3. Configure JOC for all the managed servers for a given cluster.

Enter 'y' when the script prompts whether you want to specify a cluster name, and also specify the cluster name and discover port, when prompted. This discovers all the managed servers for the given cluster and configure the JOC. The discover port is common for the entire JOC configuration across the cluster. For example:

```
Do you want to specify a cluster name (y/n) <y>
Enter Cluster Name : Spaces_Cluster
Enter Discover Port : 9988
```

Here is a walkthrough for using `configure-joc.py` for HA environments:

```
execfile('MW_HOME/oracle_common/bin/configure-joc.py')
.
Enter Hostnames (eg host1,host2) : SOAHOST1, SOAHOST2
.
Do you want to specify a cluster name (y/n) <y>y
.
Enter Cluster Name : Spaces_Cluster
.
Enter Discover Port : 9988
.
Enter Distribute Mode (true|false) <true> : true
.
Do you want to exclude any server(s) from JOC configuration (y/n) <n> n
```

The script can also be used to perform the following JOC configurations:

- Configure JOC for all specified managed servers.

Enter 'n' when the script prompts whether you want to specify a cluster name, and also specify the managed server and discover port, when prompted. For example:

```
Do you want to specify a cluster name (y/n) <y>n
Enter Managed Server and Discover Port (eg WLS_Spaces1:9988, WLS_Spaces2:9988)
: WLS_Spaces1:9988,WLS_Spaces2:9988
```

- Exclude JOC configuration for some managed servers.

The script allows you to specify the list of managed servers for which the JOC configuration "DistributeMode" will be set to 'false'. Enter 'y' when the script prompts whether you want to exclude any servers from JOC configuration, and enter the managed server names to be excluded, when prompted. For example:

```
Do you want to exclude any server(s) from JOC configuration (y/n) <n>y
Exclude Managed Server List (eg Server1,Server2) : WLS_Spaces1,WLS_Spaces3
```

- Disable the distribution mode for all managed servers.

The script allows you to disable the distribution to all the managed servers for a specified cluster. Specify 'false' when the script prompts for the distribution mode. By default, the distribution mode is set to 'true'.

Verify JOC configuration using the CacheWatcher utility. See *Oracle Fusion Middleware High Availability Guide*.

You can configure the Java Object Cache (JOC) using the **HA Power Tools** tab in the Oracle WebLogic Administration Console as described in the *Oracle Fusion Middleware High Availability Guide*.

6.14 Converting Discussions Forum from Multicast to Unicast

To convert Discussions Forum from multicast to unicast:

Step 1: Add the startup parameters

To add the relevant startup parameters:

1. In the Oracle WebLogic Server Administration Console, select **Servers, WLS_Services1, Configuration**, and then **Server Start**.
2. In the **Arguments** box, add the following:


```
-Dtangosol.coherence.wka1=WCHost1 -Dtangosol.coherence.wka2=WCHost2
-Dtangosol.coherence.localhost=WCHost1 -Dtangosol.coherence.wka1.port=8089
-Dtangosol.coherence.wka2.port=8089
```

Where **WCHost1** is where WLS_Services1 is running.

Port 8089 is a port reserved for WebCenter Coherence communications.

3. Repeat steps 1 and 2 for WLS_Services2, swapping WCHost1 for WCHost2 and WCHost2 for WCHost1.
4. Restart the WLS_Services servers.

Step 2: Validate the changes

To validate the changes:

1. Log on to the Discussions Forum Administration panel.
2. Select **Cache Settings** in the left pane.
3. At the bottom of the screen, ensure that **Clustering** is set to **enabled**.
4. Repeat steps 1 through 3 for all members of the cluster.

As servers join the cluster they appear at the top of the screen.

6.15 Configuring Clustering for Discussion Server

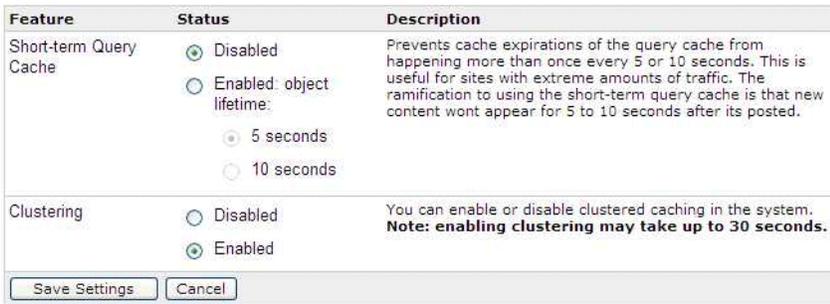
If this is a Unicast cluster, first ensure that the steps in [Section 6.14, "Converting Discussions Forum from Multicast to Unicast"](#) are performed first.

Ensure that all members of the Discussion Server cluster can communicate with each other using the Discussion Server Administration Console:

1. Login to each member of the cluster at:


```
http://<host>:<port>/owc_discussions/admin
```
2. Go to **Cache Settings**.

Figure 6–1 Cache Settings Section



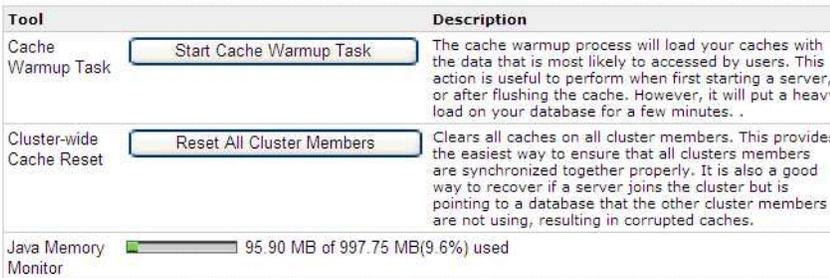
Screenshot of the Cache Settings section of the Discussion Server Administration Console.

- At the bottom of the page, in the **Cache Features** section, ensure that **Clustering** is set to **Enabled**.

The top of the page should now list all members of the cluster.

- Again, towards the end of the page, under the **Cache Tools** section, do **Cluster wide cache reset and the Cache warm up Task**. Repeat the Cache warm up task on all members of the cluster.

Figure 6–2 Cache Tools Section



Screenshot of the Cache Settings section of the Discussion Server Administration Console.

6.16 Configuring WS-Security for Discussion Server

This section describes the procedures for configuring WS-Security for the Discussion Server. These procedures are mandatory. The Client in the EDG case is WebCenter Spaces. The Server is the Discussion Server, which in this case is also on the same machine. For example, WCHOST1 and WCHOST2 function as both the client and the server machines.

6.16.1 Setting up the default Keystores

To set up the default keystores:

Note: For a cluster setup, the default/demo keystores cannot be used. They are different for each instance. You **MUST** generate a new keystore.

1. Create/get a valid signed certificate and import it into a keystore that is accessible by the client library. For example, in development, you can create and self sign a certificate by using the keytool in the JDK.

```
keytool -genkey -alias <ALIAS> -keypass <KEY_PASSWORD>
-keystore <KEY_STORE_FILE> -storepass <STORE_PASS> -dname <NAME> -keyalg RSA
```

For example:

```
keytool -genkey -alias jive -keypass jive_password -keystore
client_certs.keystore -storepass clientKeyStorePassword -dname "cn=jive"
-keyalg RSA
```

```
keytool -selfcert -alias <ALIAS> -keystore <KEY_STORE_FILE>
-storepass <CLIENT_KEY_PASSWORD> -keypass <KEY_PASSWORD>
```

For example:

```
keytool -selfcert -alias jive -keystore client_certs.keystore -storepass
clientKeyStorePassword -keypass jive_password
```

The `client_certs.keystore` is generated from the this command, use this as the client certificate.

2. Export the public key from the certificate.

```
keytool -export -alias <ALIAS> -file <CLIENT_CERTS_FILE.TYPE> -keystore <KEY_
STORE_FILE> -storepass <CLIENT_KEY_STORE_PASSWORD>
```

For example:

```
keytool -export -alias jive -file client_public_key.jks -keystore
client_certs.keystore -storepass clientKeyStorePassword
```

3. On the server, import the public key into a Web app accessible keystore.

```
keytool -import -alias <ALIAS> -file <CLIENT_CERTS_FILE.TYPE> -keystore
<SERVER_CERT_KEystore_FILE> -storepass <SERVER_KEystore_PASS>
```

For example:

```
keytool -import -alias jive -file client_public_key.jks
-keystore server_public_certs.keystore -storepass serverKeyStorePassword
```

The `server_public_certs.keystore` file is generated from this command. Use this as the server certificate.

6.16.2 Distributing the Keystores

The client keystores generated in the previous section must be distributed onto all the machines with ws clients. In an EDG setup this includes all the machines with WebCenter Spaces. The keystores can be located anywhere, as long as the path to the keystores is the same on all the machines in the cluster. For EDG Oracle recommend `/u01/app/oracle/keystores/`.

The server keystores generated in the previous section must be distributed onto all the machines running Discussion Forum. The keystores can be located anywhere, but the path to the keystores must be the same on all the machines in the cluster.

The rest of this example assumes all of the keystores are located in `/u01/app/oracle/keystores/`.

6.16.3 Set up Jive Server Properties

To configure ws-security in Jive Server:

1. Go to `DOMAIN_HOME/lib` (on any machine in the cluster).
2. Create a `keystore.properties` file, insert the following values, and save the file.

```
org.apache.ws.security.crypto.provider=org.apache.ws.security.components.crypto
.Merlin
org.apache.ws.security.crypto.merlin.keystore.type=jks
org.apache.ws.security.crypto.merlin.keystore.password=serverKeyStorePassword
org.apache.ws.security.crypto.merlin.keystore.alias=jive
org.apache.ws.security.crypto.merlin.file=/u01/app/oracle/keystores/server_
public_certs.keystore
```

The alias and password are the ones used when generating the keystore described above. The file path is the full path to the server certificate generated.

3. Place the keystore file into a jar file.

```
jar cvf jive_crypto_props.jar keystore.properties
```

Note: The jar size generated must be close to 616.

4. `rm keystore.properties.`
5. Distribute the `jive_crypto_props.jar` file to all machines in the cluster running Jive.
6. Access the Jive administration page:
`http://host:port/owc_discussions/admin`
7. Log in as `admin` user.
8. Click on the **System Properties** link, and add the following properties:
`webservices.soap.custom.crypto.fileName=keystore.properties`
9. **Restart all the Jive Servers.**

6.17 Configuring Wiki Server

As outlined in [Section 2.3.2, "Recommended Locations for the Different Directories,"](#) the Administration Server domain directory and apps directory is on a shared disk. The apps directory also contains the deployment for Wiki Server. All Wiki Servers in the cluster are sharing this same apps directory. Therefore, for all Wiki servers in the cluster to run properly, the shared storage should be mounted on all nodes.

Set the **non clustered mode** setting in **Wiki Administration, Settings**, to `false` ("Use cached data") on both of the servers.

6.18 Configuring Oracle HTTP Server for the WLS_Spacesn, WLS_Portletn, and WLS_Servicesn Managed Servers on WCHOST2

To enable Oracle HTTP Server to route to the Spaces_Cluster, Portlet_Cluster, and Services_Cluster, which contain the WLS_Spacesn, WLS_Portletn, and WLS_Servicesn managed servers, you must set the WebLogicCluster parameter to the list of nodes in the cluster.

1. Add the following lines to the *OHS_HOME/instances/ohs_instance1/config/OHS/ohs1/mod_wl_ohs.conf* file:

```
# Spaces

<Location /webcenter>
    WebLogicCluster wchost1.com:9000,wchost2.com:9000
    SetHandler weblogic-handler
</Location>

<Location /webcenterhelp>
    WebLogicCluster wchost1.com:9000,wchost2.com:9000
    SetHandler weblogic-handler
</Location>

<Location /rss>
    WebLogicCluster wchost1.com:9000,wchost2.com:9000
    SetHandler weblogic-handler
</Location>

# Portlet

<Location /portalTools>
    WebLogicCluster wchost1.com:9001,wchost2.com:9001
    SetHandler weblogic-handler
</Location>

<Location /wsrp-tools>
    WebLogicCluster wchost1.com:9001,wchost2.com:9001
    SetHandler weblogic-handler
</Location>

# WSM

<Location /wsm-pm>
    WebLogicCluster soahost1.com:7010,soahost2.com:7010
    SetHandler weblogic-handler
</Location>

# Discussions and Wiki

<Location /owc_discussions>
    WebLogicCluster wchost1.com:9002,wchost2.com:9002
    SetHandler weblogic-handler
</Location>

<Location /owc_wiki>
    WebLogicCluster wchost1.com:9002,wchost2.com:9002
    SetHandler weblogic-handler
</Location>
```

2. Add the following lines to the *httpd.conf* file (located in the same directory as the *mod_wl_ohs* file):

```
NameVirtualHost *:7777
<VirtualHost *:7777>
    ServerName https://wc.mycompany.com:443
    ServerAdmin you@your.address
    RewriteEngine On
    RewriteOptions inherit
</VirtualHost>
```

```
NameVirtualHost *:7777
<VirtualHost *:7777>
    ServerName admin.mycompany.com:80
    ServerAdmin you@your.address
    RewriteEngine On
    RewriteOptions inherit
</VirtualHost>
```

```
NameVirtualHost *:7777
<VirtualHost *:7777>
    ServerName wcinternal.mycompany.com:80
    ServerAdmin you@your.address
    RewriteEngine On
    RewriteOptions inherit
</VirtualHost>
```

3. Restart Oracle HTTP Server on both WEBHOST1 and WEBHOST2:

```
WEBHOST1> ORACLE_BASE/admin/<instance_name>/bin/opmnctl restartproc
ias-component=ohs1
WEBHOST2> ORACLE_BASE/admin/<instance_name>/bin/opmnctl restartproc
ias-component=ohs1
```

The servers specified in the `WebLogicCluster` parameter are only important at startup time for the plug-in. The list needs to provide at least one running cluster member for the plug-in to discover other members of the cluster. Note that the listed cluster member must be running when Oracle HTTP Server is started. Oracle WebLogic Server and the plug-in work together to update the server list automatically with new, failed, and recovered cluster members.

Some example scenarios:

- Example 1: If you have a two-node cluster and then add a third member, you do not need to update the configuration to add the third member. The third member will be discovered on the fly at runtime.
- Example 2: You have a three-node cluster but only two nodes are listed in the configuration. However, if both listed nodes are down when you start Oracle HTTP Server, then the plug-in would fail to route to the cluster. You must ensure that at least one of the listed nodes is running when you start Oracle HTTP Server.

If you list all members of the cluster, then you guarantee you can route to the cluster, assuming at least one member is running when Oracle HTTP Server is started.

For more information on configuring the WebLogic Server plug-in, see the Oracle Fusion Middleware Using Web Server Plug-Ins With Oracle WebLogic Server guide.

6.19 Validating Access Through Oracle HTTP Server

Verify that you can access these URLs:

- `http://webhostN:7777/webcenter`
- `http://webhostN:7777/webcenterhelp`
- `http://webhostN:7777/rss`
- `http://webhostN:7777/portalTools`
- `http://webhostN:7777/wsrp-tools`
- `http://webhostN:7777/owc_wiki`
- `http://webhostN:7777/owc_discussions`

where 'webhostN' specifies the name of each Oracle HTTP Server host (for example, WEBHOST1, WEBHOST2).

6.20 Validating Access Through the Load Balancer

Verify that you can access these URLs:

- `https://wc.mycompany.com/webcenter`
- `https://wc.mycompany.com/webcenterhelp`
- `https://wc.mycompany.com/rss`
- `https://wc.mycompany.com/portalTools`
- `https://wc.mycompany.com/wsrp-tools`
- `http://wcinternal.mycompany.com/owc_wiki`
- `http://wcinternal.mycompany.com/owc_discussions`

6.21 Backing Up the Installation

After you have verified that the extended domain is working, back up the installation. This is a quick backup for the express purpose of immediate restore in case of problems in the further steps. The backup destination is the local disk. This backup can be discarded once the enterprise deployment setup is complete. At this point, the regular deployment-specific backup and recovery process can be initiated. The *Oracle Fusion Middleware Administrator's Guide* provides further details. For information on describing the Oracle HTTP Server data that must be backed up and restored, refer to the "Backup and Recovery Recommendations for Oracle HTTP Server" section in this guide. For information on how to recover components, see "Recovery of Components" and "Recovery After Loss of Component" sections in the guide. For recommendations specific to recovering from the loss of a host, see the "Recovering Oracle HTTP Server to a Different Host" in the guide. Also refer to the *Oracle Database Backup and Recovery Guide* for information on database backup.

To back up the installation at this point, complete these steps:

1. Back up the web tier:
 - a. Shut down the instance using `opmnctl`.

```
ORACLE_BASE/admin/<instance_name>/bin/opmnctl stopall
```

- b.** Back up the Middleware Home on the web tier using the following command (as root):

```
tar -cvpf BACKUP_LOCATION/web.tar $MW_HOME
```

- c.** Back up the Instance Home on the web tier using the following command (as root):

```
tar -cvpf BACKUP_LOCATION/web_instance.tar $ORACLE_INSTANCE
```

- d.** Start the instance using `opmnctl`:

```
ORACLE_BASE/admin/<instance_name>/bin/opmnctl startall
```

- 2.** Back up the database. This is a full database backup (either hot or cold) using Oracle Recovery Manager (recommended) or OS tools such as `tar` for cold backups if possible.
- 3.** Back up the AdminServer domain directory. Perform a backup to save your domain configuration. The configuration files all exist under the `ORACLE_BASE/admin/<domain_name>` directory.

```
SOAHOST1> tar -cvpf edgdomainback.tar ORACLE_BASE/admin/<domain_name>
```

Setting Up Node Manager

This chapter describes how to configure Node Manager per the EDG recommendations. Oracle Fusion Middleware EDG recommends using host name verification for the communications between Node Manager and the Administration Server. This requires the use of certificates for the different addresses communicating with the Administration Server. In this chapter, the steps for configuring SOAHOST1 and SOAHOST2 certificates for host name verification are provided. Similar steps are required for WCHOST1 and WCHOST2. Although the appropriate host name changes in the steps are required for WCHOST1 and WCHOST2, the procedure and syntax are exactly the same.

This chapter includes the following sections:

- [Section 7.1, "About the Node Manager"](#)
- [Section 7.2, "Enabling Host Name Verification Certificates for Node Manager in SOAHOST1"](#)
- [Section 7.3, "Starting the Node Manager on SOAHOST1"](#)
- [Section 7.4, "Enabling Host Name Verification Certificates for the Node Manager in SOAHOST2"](#)
- [Section 7.5, "Starting the Node Manager on SOAHOST2"](#)
- [Section 7.6, "Enabling Host Name Verification Certificates for Node Manager in WCHOST1 and WCHOST2"](#)

7.1 About the Node Manager

The Node Manager enables you to start and stop the Administration Server and the managed servers.

About Passwords

The passwords used in this guide are used only as examples. Use secure passwords in a production environment. For example, use passwords that include both uppercase and lowercase characters as well as numbers.

7.2 Enabling Host Name Verification Certificates for Node Manager in SOAHOST1

Perform these steps to set up host name verification certificates for communication between the Node Manager and the Administration Server.

- Step 1: [Generating Self-Signed Certificates Using the `utils.CertGen` Utility](#)

- Step 2: [Creating an Identity Keystore Using the `utils.ImportPrivateKey` Utility](#)
- Step 3: [Creating a Trust Keystore Using the `Keytool` Utility](#)
- Step 4: [Configuring Node Manager to Use the Custom Keystores](#)

7.2.1 Generating Self-Signed Certificates Using the `utils.CertGen` Utility

Follow these steps to create self-signed certificates on SOAHOST1.mycompany.com. These certificates should be created using the network name/alias. For information on using trust CA certificates instead, see "Configuring Identity and Trust" in *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

1. Set up your environment by running the `WL_HOME/server/bin/setWLSEnv.sh` script:

In the Bourne shell, run the following command:

```
SOAHOST1> . setWLSEnv.sh
```

Verify that the `CLASSPATH` environment variable is set:

```
SOAHOST1> echo $CLASSPATH
```

2. Create a user-defined directory for the certificates. For example, create a directory called `certs` under the `ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>` directory. Note that certificates can be shared across WLS domains.

```
SOAHOST1> cd ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>
SOAHOST1> mkdir certs
```

3. Change directory to the user-defined directory.

```
SOAHOST1> cd certs
```

4. Run the `utils.CertGen` tool from the user-defined directory to create the certificates for both SOAHOST1 and SOAHOST1VHN1.

Syntax:

```
java utils.CertGen <key_passphrase> <cert_file_name> <key_file_name> [export | domestic] [hostname]
```

Examples:

```
SOAHOST1> java utils.CertGen welcome1 SOAHOST1_cert SOAHOST1_key
domestic SOAHOST1.mycompany.com
```

```
SOAHOST1> java utils.CertGen welcome1 VIPHOST1_cert VIPHOST1_key
domestic SOAHOST1VHN1.mycompany.com
```

7.2.2 Creating an Identity Keystore Using the `utils.ImportPrivateKey` Utility

Follow these steps to create an Identity Keystore on SOAHOST1.mycompany.com.

1. Create a new identity keystore called `appIdentityKeyStore` using the `utils.ImportPrivateKey` utility.

Create this keystore under the same directory as the certificates (that is, `ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>/certs`).

Note: The Identity Store is created (if none exists) when you import a certificate and the corresponding key into the Identity Store using the `utils.ImportPrivateKey` utility.

Import the certificate and private key for both SOAHOST1 and VIPHOST1 into the Identity Store. Make sure that you use a different alias for each of the certificate/key pair imported.

Syntax:

```
java utils.ImportPrivateKey <keystore_file> <keystore_
password> <certificate_alias_to_use> <private_key_passphrase>
<certificate_file> <private_key_file> [<keystore_type>]
```

Examples:

```
SOAHOST1> java utils.ImportPrivateKey appIdentityKeyStore.jks welcome1
appIdentity1 welcome1
ORACLE_BASE/admin/<domain_name>/aserver/<domain_
name>/certs/SOAHOST1_cert.pem
ORACLE_BASE/admin/<domain_name>/aserver/<domain_
name>/certs/SOAHOST1_key.pem
```

```
SOAHOST1> java utils.ImportPrivateKey appIdentityKeyStore.jks welcome1
appIdentity2 welcome1
ORACLE_BASE/admin/<domain_name>/aserver/<domain_
name>/certs/VIPHOST1_cert.pem
ORACLE_BASE/admin/<domain_name>/aserver/<domain_
name>/certs/VIPHOST1_key.pem
```

7.2.3 Creating a Trust Keystore Using the Keytool Utility

Follow these steps to create the Trust Keystore on SOAHOST1.mycompany.com.

1. Copy the standard java keystore to create the new trust keystore since it already contains most of the root CA certificates needed. Oracle does not recommend modifying the standard Java trust key store directly. Copy the standard Java keystore CA certificates located under the `WL_HOME/server/lib` directory to the same directory as the certificates. For example:

```
cp WL_HOME/server/lib/cacerts
ORACLE_BASE/admin/<domain_name>/aserver/<domain_
name>/certs/appTrustKeyStore.jks
```

2. The default password for the standard Java keystore is `changeit`. Oracle recommends always changing the default password. Use the keytool utility to do this. The syntax is:

```
keytool -storepasswd -new <NewPassword> -keystore <TrustKeyStore> -storepass
<Original Password>
```

For example:

```
keytool -storepasswd -new welcome1 -keystore appTrustKeyStore.jks -storepass
changeit
```

3. The CA certificate `CertGenCA.der` is used to sign all certificates generated by the `utils.CertGen` tool and is located at `WL_HOME/server/lib` directory. This CA certificate must be imported into the `appTrustKeyStore` using the keytool utility. The syntax is:

```
keytool -import -v -noprompt -trustcacerts -alias <AliasName>
-file <CAFileLocation> -keystore <KeyStoreLocation> -storepass <KeyStore
Password>
```

For example:

```
keytool -import -v -noprompt -trustcacerts -alias clientCACert -file
$WL_HOME/server/lib/CertGenCA.der -keystore appTrustKeyStore.jks -storepass
welcome1
```

7.2.4 Configuring Node Manager to Use the Custom Keystores

To configure the Node Manager to use the custom keystores, add the following lines to the end of the `nodemanager.properties` file located in the `WL_HOME/common/nodemanager` directory:

```
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityKeyStoreFileName=<Identity KeyStore>
CustomIdentityKeyStorePassPhrase=<Identity KeyStore Passwd>
CustomIdentityAlias=<Identity Key Store Alias>
CustomIdentityPrivateKeyPassPhrase=<Private Key used when creating Certificate>
Make sure to use the correct value for CustomIdentityAlias on each node. For
example on SOAHOST1, use appIdentity1, and on VIPHOST1, use appIdentity2.
Example for Node 1:
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityKeyStoreFileName=ORACLE_BASE/admin/<domain_name>/aserver/<domain_
name>/certs/appIdentityKeyStore.jks
CustomIdentityKeyStorePassPhrase=welcome1
CustomIdentityAlias=appIdentity1
CustomIdentityPrivateKeyPassPhrase=welcome1
```

The passphrase entries in the `nodemanager.properties` file get encrypted when you start Node Manager as described in [Section 7.3, "Starting the Node Manager on SOAHOST1."](#) For security reasons, you want to minimize the time the entries in the `nodemanager.properties` file are left unencrypted. After you edit the file, you should start Node Manager as soon as possible so that the entries get encrypted.

When using a common/shared storage installation for `MW_HOME`, Node Manager is started from different nodes using the same base configuration (`nodemanager.properties`). In that case, it is required to add the certificate for all the nodes that share the binaries to the `appIdentityKeyStore.jks` identity store. To do this, create the certificate for the new node and import it to `appIdentityKeyStore.jks` as described above. Once the certificates are available in the store, each node manager needs to point to a different identity alias to send the correct certificate to the Administration Server. To do this, set different environment variables before starting Node Manager in the different nodes:

```
SOAHOST1>cd WL_HOME/server/bin
SOAHOST1>export JAVA_OPTIONS=-DCustomIdentityAlias=appIdentitySOAHOST1
```

```
SOAHOSTn>cd WL_HOME/server/bin
SOAHOSTn>export JAVA_OPTIONS=-DCustomIdentityAlias=appIdentitySOAHOSTn
```

7.3 Starting the Node Manager on SOAHOST1

Run these commands to start Node Manager on SOAHOST1:

Note: If you have not configured and started Node Manager for the first time yet, run the `setNMProps.sh` script as specified in section [Section 4.11, "Starting Node Manager on SOAHOST1."](#) This will enable the use of the start script which is required for SOA.

```
SOAHOST1> cd WL_HOME/server/bin
SOAHOST1> ./startNodeManager.sh
```

7.4 Enabling Host Name Verification Certificates for the Node Manager in SOAHOST2

Perform these steps to set up SSL for communication between the Node Manager and the Administration Server:

- Step 1: [Generating Self-Signed Certificates Using the `utils.CertGen` Utility](#)
- Step 2: [Creating an Identity Keystore Using the `"utils.ImportPrivateKey"` Utility](#)
- Step 3: [Creating a Trust Keystore Using the `keytool` Utility](#)
- Step 4: [Configuring Node Manager to Use the Custom Keystores](#)

7.4.1 Generating Self-Signed Certificates Using the `utils.CertGen` Utility

Follow these steps to create self-signed certificates on SOAHOST2.mycompany.com. These certificates should be created using the network name/alias.

1. Set up your environment by running the `WL_HOME/server/bin/setWLSEnv.sh` script:

In the Bourne shell, run the following command:

```
SOAHOST2> . setWLSEnv.sh
```

Verify that the `CLASSPATH` environment variable is set:

```
SOAHOST2> echo $CLASSPATH
```

2. Create a user-defined directory for the certificates. For example, create a directory called `certs` under the `ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>` directory. Note that certificates can be shared across WLS domains.

```
SOAHOST2> cd ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>
SOAHOST2> mkdir certs
```

Note: The directory where keystores and trust keystores are maintained must be on shared storage that is visible from all nodes so that when the Administration Server fails over the cert/keystores are also available.

3. Change directory to the user-defined directory.

```
SOAHOST2> cd certs
```

4. Run the `utils.CertGen` tool from the user-defined directory to create the certificates for both SOAHOST2 and VIPHOST1.

Syntax:

```
java utils.CertGen <key_passphrase> <cert_file_name> <key_
file_name> [export | domestic] [hostname]
```

Examples:

```
SOAHOST2> java utils.CertGen welcome1 SOAHOST2_cert SOAHOST2_key
domestic SOAHOST2.mycompany.com
```

```
SOAHOST2> java utils.CertGen welcome1 VIPHOST1_cert VIPHOST1_key
domestic SOAHOST1VHN1.mycompany.com
```

7.4.2 Creating an Identity Keystore Using the "utils.ImportPrivateKey" Utility

Follow these steps to create an Identity Keystore on SOAHOST2.mycompany.com.

1. Create a new identity keystore called "appIdentityKeyStore" using the "utils.ImportPrivateKey" utility.

Create this keystore under the same directory as the certificates (that is, ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>/certs).

Note that the Identity Store is created (if none exists) when you import a certificate and the corresponding key into the Identity Store using the "utils.ImportPrivateKey" utility.

Import the certificate and private key for both SOAHOST2 and VIPHOST1 into the Identity Store. Make sure that you use a different alias for each of the certificate/key pair imported.

Syntax:

```
java utils.ImportPrivateKey <keystore_file> <keystore_
password> <certificate_alias_to_use> <private_key_passphrase>
<certificate_file> <private_key_file> [<keystore_type>]
```

Examples:

```
SOAHOST2> java utils.ImportPrivateKey appIdentityKeyStore.jks welcome1
appIdentity1 welcome1
ORACLE_BASE/admin/<domain_name>/aserver/<domain_
name>/certs/SOAHOST1_cert.pem
ORACLE_BASE/admin/<domain_name>/aserver/<domain_
name>/certs/SOAHOST1_key.pem
```

```
SOAHOST2> java utils.ImportPrivateKey appIdentityKeyStore.jks welcome1
appIdentity2 welcome1
ORACLE_BASE/admin/<domain_name>/aserver/<domain_
name>/certs/VIPHOST1_cert.pem
ORACLE_BASE/admin/<domain_name>/aserver/<domain_
name>/certs/VIPHOST1_key.pem
```

7.4.3 Creating a Trust Keystore Using the keytool Utility

Follow these steps to create the Trust Keystore on SOAHOST2.mycompany.com.

1. Create a new trust keystore called *appTrustKeyStore* using the *keytool* utility:

```
bash-3.00$ keytool -keystore appTrustKeyStore.jks -genkey -keyalg RSA
-alias app TrustKey -dname "cn=appTrustKey,ou=FOR TESTING ONLY,
o=MyOrganization,L=MyTown,ST=MyState,C=US"
Enter keystore password:
Re-enter new password:
```

```
Enter key password for <appTrustKey>
(RETURN if same as keystore password):
bash-3.00$ _
```

Note: Use the standard Java keystore to create the new trust keystore because it already contains most of the needed root CA certificates. Do not to modify the standard Java trust key store directly.

2. You will be asked a series of questions. The keystore is created after you respond to these questions.

Tip: Make a note of the information that you provide on the command line and in the subsequent dialog box, because you will need this information to define gateway policy steps.

3. Change the default password for the standard Java keystore utility using the `keytool` utility. Use the following syntax to change the default password:

```
keytool -storepasswd -keystore <TrustKeyStore>
```

4. Import the CA certificate called `CertGenCA.der` into the `appTrustKeyStore` using the `keytool` utility. This certificate, which is located in the `WL_HOME/server/lib` directory, is used to sign all certificates generated by `utils.CertGen` tool. Import `CertGenCA.der` using the following syntax:

```
keytool -import -v -noprompt -trustcacerts -alias <AliasName>
-file <CAFileLocation> -keystore <KeyStoreLocation>
```

7.4.4 Configuring Node Manager to Use the Custom Keystores

Follow these steps to configure the Node Manager to use the custom keystores.

1. Add the following lines to the end of the `nodemanager.properties` file located in the `WL_HOME/common/nodemanager` directory.

```
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityKeyStoreFileName=<Identity KeyStore>
CustomIdentityKeyStorePassPhrase=<Identity KeyStore Passwd>
CustomIdentityAlias=<Identity Key Store Alias>
CustomIdentityPrivateKeyPassPhrase=<Private Key used when
creating Certificate>
```

Make sure to use the correct value for `CustomIdentityAlias` on each node. For example on SOAHOST2, use "appIdentity2", and on VIPHOST1, use "appIdentity2".

Example for Node 1:

```
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityKeyStoreFileName=ORACLE_BASE/admin/<domain_
name>/aserver/<domain_name>/certs/appIdentityKeyStore.jks
CustomIdentityKeyStorePassPhrase=welcome1
CustomIdentityAlias=appIdentity1
CustomIdentityPrivateKeyPassPhrase=welcome1
```

Note: The passphrase entries in the `nodemanager.properties` file get encrypted when you start Node Manager, as described in [Section 7.5, "Starting the Node Manager on SOAHOST2."](#)

For security reasons, you want to minimize the time the entries in the `nodemanager.properties` file are left unencrypted. After you edit the file, you should start Node Manager as soon as possible so that the entries get encrypted.

7.5 Starting the Node Manager on SOAHOST2

Run these commands to start Node Manager on SOAHOST2:

Note: If you have not configured and started Node Manager for the first time yet, run the `setNMProps.sh` script as specified in [Section 4.15, "Starting Node Manager on SOAHOST2."](#) This will enable the use of the start script which is required for SOA.

```
SOAHOST2> cd WL_HOME/server/bin
SOAHOST2> ./startNodeManager.sh
```

7.6 Enabling Host Name Verification Certificates for Node Manager in WCHOST1 and WCHOST2

Repeat the steps in [Section 7.4, "Enabling Host Name Verification Certificates for the Node Manager in SOAHOST2,"](#) and [Section 7.5, "Starting the Node Manager on SOAHOST2,"](#) substituting WCHOST1 and WCHOST2 for SOAHOST2, to configure SSL for the node managers on these machines.

Configuring External Services

Configuring external services involves establishing a connection from the WebLogic domain to the location of the external service. This chapter describes how to configure the services using WebCenter Spaces or the WebLogic scripting tool (WLST).

This chapter contains the following sections:

- [Section 8.1, "Configuring the Discussion Forum Connection"](#)
- [Section 8.2, "Configuring the Instant Messaging and Presence \(IMP\) Server Connection"](#)
- [Section 8.3, "Configuring the Worklist and Workflow Server Connection"](#)
- [Section 8.4, "Configuring the Document Library Connection"](#)
- [Section 8.5, "Registering the Portlet Producers"](#)

8.1 Configuring the Discussion Forum Connection

The discussion forum connection can be configured using either WebCenter Spaces or via the WebLogic Scripting Tool (WLST):

- [Section 8.1.1, "Configuring Discussions using WebCenter Spaces"](#)
- [Section 8.1.2, "Configuring Discussions via WLST"](#)

8.1.1 Configuring Discussions using WebCenter Spaces

For details on configuring discussions using WebCenter Spaces, see section 12.1.3 of the *WebCenter Administration Guide*. The required values for the WebCenter EDG configuration are as follows:

- Server URL: **http://wcinternal.mycompany.com/owc_discussions**
- Admin User: **the discussion server admin user name**
- Admin Password: **the discussion server admin password**

8.1.2 Configuring Discussions via WLST

To configure discussions via the WebLogic Scripting Tool, complete these steps:

1. Start the WebLogic Scripting Tool:

```
WCHOST1> MW_HOME/wc/common/bin/wlst.sh
```

2. In WLST, connect as the administrator:

```
> connect('weblogic','<admin password>','SOAHOST1VHN1:7001')
```

- At the WebCenter App side, create the forum connection as follows, using WLST.

Note: Note that the alias and passwords are those that were used in [Section 6.16, "Configuring WS-Security for Discussion Server,"](#) when generating the keystore.

```
createDiscussionForumConnection(appName='webcenter',name='Jive-DiscussionForum',url=DF_URL,adminUser=DF_USER,secured=true,default=true)

setDiscussionForumConnectionProperty('webcenter','Jive-DiscussionForum',key='keystore.location',value='/u01/app/oracle/keystores/jive_client_certs.keystore')

setDiscussionForumConnectionProperty('webcenter','Jive-DiscussionForum',key='keystore.password',value='clientKeyStorePassword')

setDiscussionForumConnectionProperty('webcenter','Jive-DiscussionForum',key='keystore.type',value='jks')

setDiscussionForumConnectionProperty('webcenter','Jive-DiscussionForum',key='encryption.key.alias',value='jive')

setDiscussionForumConnectionProperty('webcenter','Jive-DiscussionForum',key='encryption.key.password',value='jive_password')
```

8.1.3 Creating a Discussions Server Connection for WebCenter From EM

Create a Discussions Server connection for WebCenter from the EM:

- Log on to an EM instance, register your WebCenter domain.
- Log on to this WebCenter domain.
- From the WebCenter Spaces **WebCenter** drop-down menu, select **Settings**, and then **Service Configuration**.
- Click **Discussions and Announcements**, and then on **Add**.
- In the Add Discussion and Announcement Connection screen, enter the name (DFConnection), service URL (http://<host>:<port>/owc_discussions) and administrator user name.
- Click **OK** to save the settings.
- Restart the WLS_Spaces managed server.

Once you log in to WebCenter, you are be SSO'ed to the Discussion server as well.

8.2 Configuring the Instant Messaging and Presence (IMP) Server Connection

The Instant Messaging and Presence (IMP) connection can be configured using either WebCenter Spaces or via the WebLogic Scripting Tool (WLST):

- [Section 8.2.1, "Configuring IMP using WebCenter Spaces"](#)
- [Section 8.2.2, "Configuring IMP via WLST"](#)

8.2.1 Configuring IMP using WebCenter Spaces

For details on configuring Instant Messaging and Presence using WebCenter Spaces, see section 11.2.3, *Registering Instant Messaging and Presence Servers*, of the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter*. The required values for the WebCenter EDG configuration are as follows:

- Connection Type: **OWLSC**
- Domain: **the domain of the presence server configured when installed**
- Server URL: **http://LBR_SVC:8011/**

8.2.2 Configuring IMP via WLST

To configure IMP via the WebLogic Scripting Tool, complete these steps:

1. Start the WebLogic Scripting Tool:

```
WCHOST1> MW_HOME/wc/common/bin/wlst.sh
```

2. In WLST, connect as the administrator:

```
> connect('weblogic','<admin password>','SOAHOST1VHN1:7001')
```

3. Run the following commands in WLST to configure the IMP connection (please note that the URLs and domains are examples; you should substitute them with values appropriate to your Presence Server installation):

```
> createIMPConnection(appName='webcenter', name='IMPService-OCMS',
url='http://LBR_SVC:8011/', adapter='OCMS', domain='mydomain.com',
default=true, server='WLS_Spaces1')
> listIMPConnections(appName='webcenter', verbose=false)
```

8.3 Configuring the Worklist and Workflow Server Connection

Before configuring the connection to the BPEL Server which will host the Worklist and Workflow application, there are several prerequisites that need to be met:

1. The backend requirements for WebCenter Spaces workflows must be completed. This includes the deployment of WebCenter Spaces workflows. These steps can be found in the *Oracle WebCenter Installation and Configuration Guide*.
2. The keystores for the WebCenter and SOA servers must be generated. In particular, the steps for configurations when SOA and WebCenter are installed in the same domain must be followed. These steps can be found in the *Oracle WebCenter Installation and Configuration Guide*.

The Worklist Server connection can be configured using either WebCenter Spaces or via the WebLogic Scripting Tool (WLST):

- [Section 8.3.1, "Configuring Worklist and Workflow via WebCenter Spaces"](#)
- [Section 8.3.2, "Configuring Worklist and Workflow via WLST"](#)

8.3.1 Configuring Worklist and Workflow via WebCenter Spaces

For details on configuring Worklist via WebCenter Spaces, see section 12.5.2.2 of the *WebCenter Administration Guide*. The required values for the WebCenter EDG configuration are as follows:

- SOAP Server URL: **http://soainternal.mycompany.com**

For details on configuring Workflow, please ensure the prerequisites have been met according to section 4.7 of the *Oracle Fusion Middleware Installation Guide for Oracle WebCenter 11g Release 1 (11.1.1)*. In particular, ensure that WebCenter Spaces Workflow has been deployed on the SOA Server.

8.3.2 Configuring Worklist and Workflow via WLST

To configure Worklist via the WebLogic Scripting Tool, complete these steps:

1. Start the WebLogic Scripting Tool:

```
WCHOST1> MW_HOME/wc/common/bin/wlst.sh
```

2. In WLST, connect as the administrator.

```
> connect('weblogic','<admin password>','SOAHOST1VHN1:7001')
```

3. Run the following commands in WLST to configure the worklist server connection:

```
> createBPPELConnection('webcenter', 'WebCenter-Worklist',  
'http://soainternal.mycompany.com')  
> addWorklistConnection('webcenter', 'WebCenter-Worklist', true)  
> listWorklistConnections('webcenter', false)
```

4. Run the following commands in WLST to configure this same connection as the workflow connection:

```
> setSpacesWorkflowConnectionName('webcenter', 'WebCenter-Worklist')  
> getSpacesWorkflowConnectionName('webcenter')
```

8.4 Configuring the Document Library Connection

The Document Library connection can be configured using either WebCenter Spaces or via the WebLogic Scripting Tool (WLST):

- [Section 8.4.1, "Configuring the Document Library via WebCenter Spaces"](#)
- [Section 8.4.2, "Configuring the Document Library via WLST"](#)

Note: This section assumes that Oracle Content Server has been installed in accordance with the instructions in [Chapter 9, "Installing and Configuring Oracle Content Server."](#)

8.4.1 Configuring the Document Library via WebCenter Spaces

For details on configuring the document library via WebCenter Spaces, see Section 11.3 of the *WebCenter Administration Guide*. The required values for the WebCenter EDG configuration are as follows:

- Repository Type: **Oracle Content Server**
- CIS Type: **Socket**
- Server Host: **wcinternal.mycompany.com**
- Server Port: **9054**

8.4.2 Configuring the Document Library via WLST

To configure the document library via the WebLogic Scripting Tool, complete these steps:

1. Start the WebLogic Scripting Tool:

```
WCHOST1> MW_HOME/wc/common/bin/wlst.sh
```

2. In WLST, connect as the administrator.

```
> connect('weblogic','<admin password>','SOAHOST1VHN1:7001')
```

3. Run the following commands in WLST to configure the document library server connection:

```
> createJCRContentServerConnection('webcenter', 'ucminstall',
socketType='socket', serverHost='ucm.mycompany.com', serverPort='9054',
isPrimary='true')
> setDocumentsSpacesProperties('webcenter', spacesRoot='<SPACES_ROOT>',
applicationName='<APPNAME>', adminUserName='sysadmin')
> listJCRContentServerConnections('webcenter',false)
```

8.5 Registering the Portlet Producers

The portlet producers can be configured using either WebCenter Spaces or via the WebLogic Scripting Tool (WLST):

- [Section 8.5.1, "Configuring the Portlet Producers via WebCenter Spaces"](#)
- [Section 8.5.2, "Configuring the Portlet Producers via WLST"](#)

8.5.1 Configuring the Portlet Producers via WebCenter Spaces

For details on configuring the portlet producers via WebCenter Spaces, see sections 13.2 and 13.3 of the *WebCenter Administration Guide*. The required values for the WebCenter EDG configuration are as follows:

- WSRP WSDL URL:
`http://wcinternal.mycompany.com/wsrp-tools/portlets/wsrp2?WSDL`
- WebClipping URL:
`http://wcinternal.mycompany.com/portalTools/webClipping/providers`
- OmniPortlet URL:
`http://wcinternal.mycompany.com/portalTools/omniPortlet/providers`

8.5.2 Configuring the Portlet Producers via WLST

To configure the portlet producers via the WebLogic Scripting Tool, complete these steps:

1. Start the WebLogic Scripting Tool:

```
WCHOST1> MW_HOME/wc/common/bin/wlst.sh
```

2. In WLST, connect as the administrator.

```
> connect('weblogic','<admin password>','SOAHOST1VHN1:7001')
```

3. Run the following commands in WLST to configure the portlet producers:

```
> registerPDKJavaProducer('webcenter', 'wc-WebClipping1',
```

```
'http://wcinternal.mycompany.com/portalTools/webClipping/providers',
serviceId='webClipping', timeout=500, establishSession=true)
> listPDKJavaProducers('webcenter','wc-WebClipping1')
> registerPDKJavaProducer('webcenter', 'wc-OmniPortlet1',
'http://wcinternal.mycompany.com/portalTools/omniPortlet/providers',
serviceId='omniPortlet', timeout=500, establishSession=true)
> listPDKJavaProducers('webcenter',name='wc-OmniPortlet1')
> registerWSRPProducer('webcenter', 'wc-WSRPTools1',
'http://wcinternal.mycompany.com/wsrp-tools/portlets/wsrp2?WSDL', timeout=500)
> listWSRPProducers('webcenter', 'wc-WSRPTools1')
```

Installing and Configuring Oracle Content Server

This chapter describes how to install and configure Oracle Content Server (OCS). It contains the following sections:

- Section 9.1, "Prerequisites"
- Section 9.2, "Installing Oracle Content Server"
- Section 9.4, "Configuring Oracle Content Server for Remote Oracle HTTP Server"
- Section 9.5, "Validating the Installation"
- Section 9.6, "Backing Up the Installation"
- Section 9.7, "Configuring Oracle Content Server for WebCenter"
- Section 9.8, "Adding Another Oracle HTTP Server and Configuring a Load Balancer"
- Section 9.9, "Reconfiguring the Content Server to Support RAC"
- Section 9.10, "Backing Up the Installation"

9.1 Prerequisites

Ensure that you meet the following prerequisites:

- The OCS schema `<prefix>_OCSEVER`. exists in the database. The schema was installed when you ran RCU. You must run RCU to install schemas needed by Oracle WebCenter.
- A shared drive is available to both of the machines (WCHOST1 and WCHOST2) on which OCS will be installed.

9.2 Installing Oracle Content Server

Run the installer from WCHOST1, which is one of the two machines that will eventually form the OCS cluster. You will install OCS on the shared disk.

Run the OCS installer from the `/install` directory on the OCS media.

```
/install/UCM/ContentServer/linux/setup.sh
```

The installer is text-based. The following lines show an installer session. Your responses are in bold.

```
Please select your locale from the list.
```

```
*4. English-US
```

```

Select installation type from the list.                *1. Install new server

Please enter the full pathname to the installation directory.
Content Server Core Folder [/oracle/ucm/server]:      /shared/oracle/ucm/server
// This is an example. Make sure that you specify an installation directory that
is on a shared drive.

Create Directory                                     *1. Yes

Java virtual machine                                 *1. Sun Java 1.5.0_11 JDK

Enter the location of the native file repository. This directory contains the
native files checked in by contributors.
Content Server Native Vault Folder [/shared/oracle/ucm/server/vault/]: < accept
default >

Create Directory                                     *1. yes

Enter the location of the web-viewable file repository. This directory contains
files that can be accessed through the web server.
Content Server Weblayout Folder [/shared/oracle/ucm/server/weblayout/]: < accept
default >

Create Directory                                     *1. yes

This server can be configured to manage its own authentication or to allow another
master to act as an authentication proxy.
Configure this server as a master or proxied server.
*1. Configure as a master server.

During installation, an admin server can be installed and configured to manage
this server. If there is already an admin server on this system, you can have the
installer configure it to administrate this server instead.
Select admin server configuration.
*1. Install an admin server to manage this server.

Enter the location of an executable to start your web browser. This browser will
be used to display the online help.
Web Browser Path [/usr/local/bin/firefox]:           < accept default >

Content Server System locale
*4. English-US

Please select the region for your timezone from the list.
*1. Use the timezone setting for your operating system

Please enter the port number that will be used to connect to the Content Server.
This port must be otherwise unused.
Content Server Port [4444]:                          9054

Please enter the port number that will be used to connect to the Admin Server.
This port must be otherwise unused.
Admin Server Port [4440]:                            9050

// Do NOT take the default. Specify the list of all the IPs of WebCenter and
// Oracle HTTP Server hosts in the cluster. For the EDG topology, this includes
// WEBHOST1, WEBHOST2, WCHOST1, WCHOST2.
// For example, 111.222.100.101|111.222.100.102|111.222.100.103|111.222.100.104
Enter a security filter for the server port. Hosts which are allowed to

```

communicate directly with the server port may access any resources managed by the server. Insure that hosts which need access are included in the filter. See the installation guide for more details.
Incoming connection address filter [127.0.0.1]: **<list of IP addresses of all the Oracle HTTP Server and WebCenter servers in the cluster>**

// Either take the default or specify a new context root
// Here we'll accept the default of 'idc'.
Web Server Relative Root [/idc/]: **/idc/**

Enter the name of the local mail server. The server will contact this system to deliver email.

Company Mail Server [mail]: **< accept default or enter name of local mail server>**

Enter the e-mail address for the system administrator.

Administrator E-Mail Address [sysadmin@mail]: **< accept default or enter email of the system administrator >**

// Next question do NOT take the DEFAULT, specify host:port which your webserver is running

Web Server HTTP Address [webhost1]: **webhost1.us.oracle.com:7777**

Enter the name for this instance. This name should be unique across your entire enterprise. It may not contain characters other than letters, numbers, and underscores.

Server Instance Name [idc]: **< accept default >**

Enter a short label for this instance. This label is used on web pages to identify this instance. It should be less than 12 characters long.

Server Instance Label [idc]: **< accept default >**

Enter a long description for this instance.

Server Description [Content Server idc]: **< accept default >**

// Choose Apache for Oracle HTTP Server

Web Server

***1. Apache**

Please select a database from the list below to use with the Content Server.

Content Server Database

***1. Oracle**

Manually configure JDBC settings for this database

***2. no**

// Specify the hostname of one of the RAC DB machines.

Oracle Server Hostname [localhost]: **rachost1.us.oracle.com**

// Specify the database port

Oracle Listener Port Number [1521]: **1521**

// Specify the username (prefix_OCSERVER) of the schemas you installed using the RCU

The user name is used to log into the database used by the content server.

Oracle User [user]: **WCEDG_OCSERVER**

// Specify the password (prefix_OCSERVER) of the schemas you installed using the RCU

The password is used to log into the database used by the content server.

```

Oracle Password []:  < enter the schema password >

//Specify the SID of one of the RAC instances.
Oracle Instance Name [ORACLE]:  orasid1

Configure the JVM to find the JDBC driver in a specific jar file
      *2. no

// Accept the default. Do not choose to create database tables since they should
// have been created with RCU already.

```

The installer can attempt to create the database tables, or you can manually create them. If you choose to manually create the tables, you should create them now. Attempt to create database tables

2. No

***1(Proceed)**

Select components to install.

1. ContentFolios: Collect related items in folios
2. LinkManager8: Hypertext link management support
3. OracleTextSearch: External Oracle 11g database as search indexer support
4. ThreadedDiscussions: Threaded discussion management

Enter numbers separated by commas to toggle, 0 to unselect all, F to finish: **F**

The installer then installs Oracle Content Server. Ensure that Oracle Content Server starts up at the end of installation.

If Oracle Content Server does not start, check the logs for errors, and resolve them. The log files are located in the following places:

Table 9–1 Location of Log Files for Oracle Content Server

Location	Description
<content_server_dir>/install/log.txt	Basic installer log
<content_server_dir>/server/etc/log	Output from starting Oracle Content Server
<content_server_dir>/weblayout/groups/secure/logs/IdcLnLog.htm	HTML logs

You can try to start the server manually by running:

```

<content_server_dir>/etc/idcserver_start

<content_server_dir>/admin/etc/idcadmin_start

```

If you want to see some debug output, specify the `-debug` parameter:

```

<content_server_dir>/bin/IdcServer -debug

```

If you want to see more debug output, edit the `<content_server_dir>/config/config.cfg` file to include the following line:

```

TraceSectionsList=*

```

If you want to see even more debug output, add these lines to `<content_server_dir>/config/config.cfg`:

```

TraceSectionsList=*
TraceIsVerbose=1

```

9.3 Running the WebCenter Configuration Script

Run the WebCenter configuration script, *wc_contentserverconfig.sh*, from the Oracle Content Server 10.1.3.4.1 media to prepare the server to work with Oracle WebCenter. This script installs the following:

- Oracle Content Server 10.1.3.4.1 patch
- Folders_g 10.1.3.4.1 component
- WcConfigure component

The WcConfigure component performs the following tasks to configure Oracle Content Server for Oracle WebCenter:

- Sets configuration values for UseAccounts and IsAutoNumber to true and AutoNumberPrefix to IDC_Name, if not set.
- Updates, if necessary, the JDBC password and its encoding from ClearText to Intradoc.
- Adds the document type DOCUMENT.
- Configures folders so that dDocType and dSecurityGroup are inherited, and the system default information is set as follows: dDocType=DOCUMENT and dSecurityGroup=Public.

Perform these steps to run the *wc_contentserverconfig* WebCenter script:

1. Navigate to the *webcenter-conf* directory, which is in the root directory on the Oracle Content Server 10.1.3.4.1 media.
2. Run the following command:

```
./wc_contentserverconfig.sh <content_server_dir> <path_to_source_directory>
```

Where *<content_server_dir>* refers to the Oracle Content Server 10.1.3.3.3 installation directory and *<path_to_source_directory>* refers to the *webcenter-conf* directory on the Oracle Content Server 10.1.3.4.1 media. For example:

```
wc_contentserverconfig.sh /shared/oracle/ucm/server /myproducts/ucmedia/  
webcenter-conf
```

3. Restart Oracle Content Server Administration Server and Oracle Content Server.

To restart Administration Server, run *<content_server_dir>/admin/etc/idcadmin_restart*.

To restart Oracle Content Server, run *<content_server_dir>/etc/idcserver_restart*.

Where *<content_server_dir>* refers to the Oracle Content Server installation directory.

Note: For more information about how to start, stop, or restart Oracle Content Server, see the *Content Server Installation Guide for UNIX* at http://download.oracle.com/docs/cd/E10316_01/owc.htm.

9.4 Configuring Oracle Content Server for Remote Oracle HTTP Server

Perform these steps to configure Oracle Content Server for remote Oracle HTTP Server (that is, Content Server and Oracle HTTP Server run on different hosts). You must perform steps on the Content Server machine and on the Oracle HTTP Server machine:

- [Section 9.4.1, "Steps to Perform on the Content Server Machine"](#)
- [Section 9.4.2, "Steps to Perform on the Oracle HTTP Server Machine"](#)

9.4.1 Steps to Perform on the Content Server Machine

Perform these steps on the Content Server machine (WCHOST1):

1. Shut down Oracle Content Server and the Content Server Administration Server:

To stop Oracle Content Server, run this command:

```
WCHOST1> <content_server_dir>/etc/idcserver_stop
```

To stop the Content Server Administration Server, run this command:

```
WCHOST1> <content_server_dir>/admin/etc/idcadmin_stop
```

2. Edit the `<content_server_dir>/config/config.cfg` file to include the following lines:

```
IntradocServerHostName=<Content_Server_HostName>
IdcAdminServerHostName=<Content_Server_HostName>
HttpServerAddress=<Web_Server_URL>
```

Also, add `SocketHostAddressSecurityFilter` if it is not already there. This should be set to a pipe-delimited list of hosts allowed to access Oracle Content Server.

Example:

```
IntradocServerHostName=wchost1.us.oracle.com
IdcAdminServerHostName=wchost1.us.oracle.com
HttpServerAddress=WEBHOST1.us.oracle.com:7777
SocketHostAddressSecurityFilter=127.0.0.1 | IP_of_WEBHOST1 | IP_of_WEBHOST2 |
IP_of_WCHOST1 | IP_of_WCHOST2
```

3. Edit the `<content_server_dir>/admin/bin/intradoc.cfg` file to include the `SocketHostAddressSecurityFilter` entry if it is not already there. See the previous step for the values for this parameter.
4. **Restart the content server.**

If you do not restart the content server, the `SecurityInfo.hda` file under `<content_server_dir>/data/users` is not updated.

9.4.2 Steps to Perform on the Oracle HTTP Server Machine

Perform these steps on the Oracle HTTP Server machine (WEBHOST1):

1. Create a local directory on WEBHOST1 with the same path name as the shared directory on WCHOST1:

```
WEBHOST1> mkdir -p /shared/oracle/ucm/server
```

The remaining steps will refer to this directory as `<cs_on_webhost_dir>`.

This is a local directory: the WEBHOST machine does not have access to the shared drive.

2. Copy the following directories (the content server stub) from the Oracle Content Server machine to the directory on WEBHOST1 created in the previous step:
 - <content_server_dir>/data
 - <content_server_dir>/idcplg (if it exists)
 - <content_server_dir>/weblayout (without the groups subdirectory)
3. On WEBHOST1, create the <cs_on_webhost_dir>/shared/os/<OS_Name>/lib directory.

For example:

```
WEBHOST1> mkdir /shared/oracle/ucm/server/shared/os/linux/lib
```

4. Copy the IdcApache22Auth.so and RedirectUrls.so files on the Content Server machine (these files are located in the <content_server_dir>/shared/os/<OS_Name>/lib directory) to the directory created in the previous step, on the WEBHOST1 machine.
5. On WEBHOST1, create the <cs_on_webhost_dir>/data/users/SystemFilters.hda file, and add the following lines:

```
@Properties LocalData
EnableRedirectUrls=1
@end
@ResultSet IdcAuthPlugins
2
iapExportedSymbolName
iapFileNameRoot
RedirectUrls
RedirectUrls
@end
```

6. Create virtual directories for Oracle HTTP Server.

Add the following lines to Oracle HTTP Server's httpd.conf file:

Note: The lines below use idc because it was the context root specified during the installation. If you specified a different context root, then enter your context root instead of idc.

```
LoadModule IdcApacheAuth <cs_on_webhost_
dir>/shared/os/linux/lib/IdcApache22Auth.so

IdcUserDB idc "<cs_on_webhost_dir>/data/users/userdb.txt"

Alias /idc "<cs_on_webhost_dir>/weblayout"
<Location /idc>
    Order allow,deny
    Allow from all
    DirectoryIndex portal.htm
    IdcSecurity idc
</Location>

UseCanonicalName Off
```

7. Restart Oracle Content Server.
8. Restart Oracle HTTP Server.

9.5 Validating the Installation

Verify that you can access the Admin interface at `http://webhost1:7777/idc`. The login is `sysadmin/idc`.

If the Admin interface does not come up, diagnose and resolve any problems before proceeding.

9.6 Backing Up the Installation

Back up all the installation files on both the Web machine and the Content Server machine.

For the HTTP Server machine, back up the configuration directory:

```
WEBHOST1> tar -cvpf ucmWHconfigback.tar /shared/oracle/ucm/server
```

For the Content Server machine, back up all install files and configuration:

```
WCHOST1> tar -cvpf ucmCSback.tar /shared/oracle/ucm/server
```

9.7 Configuring Oracle Content Server for WebCenter

To configure Oracle Content Server to work with WebCenter, perform the steps described in these sections:

- [Section 9.7.1, "Configuring Oracle Content Server"](#)
- [Section 9.7.2, "Configuring the Load Balancer"](#)
- [Section 9.7.3, "Installing Content Server Cluster"](#)

9.7.1 Configuring Oracle Content Server

After installing Oracle Content Server, you must configure the server to use the same LDAP-based identity store that Oracle WebCenter has been configured to use. You can optionally configure Oracle Content Server for using WS-Security and enabling full-text search and index. [Table 9–1](#) lists the various tasks and whether these tasks are mandatory or optional.

Table 9–2 Oracle WebCenter-Specific Post-Installation Configuration Tasks for Oracle Content Server

Task	Mandatory/Optional
Configuring the Identity Store	Mandatory
Enabling Full-Text Searching and Indexing	Optional
Configuring Secure Socket Layer (SSL) in Oracle Content Server	Optional

9.7.1.1 Configuring the Identity Store

Both Oracle Content Server and Oracle WebCenter must be configured to use the same LDAP-based identity store. By default, Oracle Content Server is not set up with an LDAP-based identity store.

To configure Oracle Content Server to use the LDAP-based identity store:

1. Start the Oracle Content Server console and log on to the server with administrative permission.
2. From the **Administration** menu, select **Providers**.
3. In the **Create a New Provider** section, click **Add** for the **ldapuser** provider type (Figure 9–1)

Figure 9–1 Creating a New Provider

Create a New Provider		
Provider Type	Description	Action
outgoing	Configuring an outgoing provider.	Add
database	Configuring a database provider.	Add
incoming	Configuring an incoming provider.	Add
preview	Configuring a preview provider.	Add
ldapuser	Configuring an LDAP user provider.	Add
httpoutgoing	Configuring an HTTP outgoing provider.	Add

4. Specify details for the LDAP provider. You must specify the following information: provider name, provider description, provider class, source path, LDAP server, LDAP suffix, and LDAP port (Figure 9–2). The LDAP server details must be of the server that Oracle WebCenter is configured to use.
 - **Provider Name** field: The name of the provider, which becomes a subdirectory in the <install_dir>/data/providers/ directory.
Example: OIDProvider
 - **Provider Description** field: User-friendly description of the provider.
Example: MyOracleOIDProvider
 - **Provider Class** field: The Java class that implements the provider.
The default is intradoc.provider.LdapUserProvider.
 - **Connection Class** field: The Java class that implements the connection to the LDAP server.
The default is intradoc.provider.LdapConnection.
 - **Configuration Class** field: The name of a Java class that performs some extra configuration. This class is useful for database providers, where the connection classes are already providers. Optional.
 - **Source Path** field: A unique string that identifies the LDAP provider. The first time a user requests credentials through the provider, this string is stored with the user information so it can be used to match the user with the provider next time the user asks for credentials. We suggest using the name of the provider as the Source Path.
Example: OIDUCMAccounts
 - **LDAP Server** field: The host name of the LDAP server.
Example: idm.mycompany.com
 - **LDAP Suffix** field: The root suffix (naming context) to use for all LDAP operations (such as dc=webcenter, dc=au, dc=oracle,dc=com). All mapping of LDAP groups to content server roles and accounts begin at this root.
Example: dc=oracle,dc=com

Note: Do not include spaces before or after commas.

- **LDAP Port** field: The port the LDAP server listens on. The default is 389.

You may also specify the LDAP admin user and password.

Note: Set the **Default Network Account** field to #none. Do not set any default role because all user security information is stored using the extended user attribute component of Oracle Content Server. You can set the **Role Prefix** and **Account Prefix** fields to any path that does not exist in the LDAP server.

Figure 9–2 Specifying Details of a New LDAP Provider

Add LDAP Provider	
Provider Name	<input type="text"/>
Provider Description	<input type="text"/>
Provider Class	intradoc.provider.LdapUserProvider
Connection Class	intradoc.provider.LdapConnection
Configuration Class	<input type="text"/>
Source Path	<input type="text"/>
LDAP Server	<input type="text"/>
LDAP Suffix	<input type="text"/>
LDAP Port	<input type="text" value="389"/>
Number of connections	<input type="text" value="5"/>
Connection timeout	<input type="text" value="10"/>
Priority	<input type="text" value="1"/>
Credential Map	<input type="text"/>
Use Netscape SDK	<input checked="" type="checkbox"/>
Use SSL	<input type="checkbox"/>
Use Group Filtering	<input type="checkbox"/>
Use Full Group Names	<input type="checkbox"/>
Account Permissions Delimiter	<input type="text" value="-"/>
Default Network Roles	<input type="text"/>
Default Network Accounts	<input type="text" value="#none"/>
Role Prefix	<input type="text" value="OU=Roles,OU=UCM,OU="/>
Depth	<input type="text" value="1"/>
<input type="button" value="Add"/>	

5. Click the **Add** button to add the LDAP provider.
6. Click the **Test** link on the main providers page to verify that the new LDAP connection works fine.

9.7.1.2 Enabling Full-Text Searching and Indexing

By default, the database used with Oracle Content Server is set up to provide metadata-only searching and indexing capabilities. You can additionally configure SQLServer, Oracle, and DB2 to support full-text searching and indexing. Configuring full-text search is optional, but advisable.

For information about enabling full-text searching and indexing, see the "Setting Up Database Search and Indexing" appendix in the Oracle Universal Content Management Content Server Installation Guide for Windows or UNIX available here:

http://download.oracle.com/docs/cd/E10316_01/ouc.htm

9.7.1.3 Configuring Secure Socket Layer (SSL) in Oracle Content Server

If Oracle Content Server and the WebCenter application in which you have created a repository connection are not on the same machine or same trusted private network, then identity propagation is not secure. To ensure its security, you must configure SSL on Oracle Content Server.

To configure SSL on Oracle Content Server, you must perform the following tasks:

- [Configuring a Keystore and Key on the Client Side](#)
- [Configuring a Keystore and Key on the Server Side](#)
- [Verifying Signatures of Trusted Clients](#)
- [Securing Identity Propagation](#)

You can also see the "Secure Socket Layer (SSL) Communication" in *Content Integration Suite (CIS) Installation Guide* available at

http://download.oracle.com/docs/cd/E10316_01/owc.htm. Perform these procedures, if you use self-signed certificates.

In a production environment, it is recommended that you use real certificates. For information on how to configure keystores when using real certificates, see "Using Service Providers" in *Security Providers Component Administration Guide* available at http://download.oracle.com/docs/cd/E10316_01/cs/cs_doc_10/documentation/extras/security_providers_guide.pdf.

Configuring a Keystore and Key on the Client Side

To configure a keystore on the WebCenter application (client) side:

1. In your development environment, go to `JDEV_HOME/jdk/bin` and open the command prompt.

2. Generate the client keystore by running the following keytool command:

```
keytool -genkey -keyalg RSA -validity 5000 -alias Client private key alias
-keystore client-keystore.jks
-dname "cn=client" -keypass Private key password -storepass KeyStore password
```

3. To verify that the keys have been correctly created, run the following keytool command. This is an optional step:

```
keytool -list -keystore client-keystore.jks -storepass KeyStore password
```

4. To use the key, sign it by running the following keytool command:

```
keytool -selfcert -validity 5000 -alias Client private key alias -keystore
client-keystore.jks
-keypass Private key password -storepass KeyStore password
```

5. Export the client public key by running the following keytool command:

```
keytool -export -alias Client private key alias -keystore client-keystore.jks
-file client.pubkey -keypass Private key password -storepass KeyStore password
```

Configuring a Keystore and Key on the Server Side

To configure a keystore on the Oracle Content Server side:

1. In the same development environment, go to *JDEV_HOME/jdk/bin* and open the command prompt.

2. Generate the server keystore by running the following keytool command:

```
keytool -genkey -keyalg RSA -validity 5000 -alias Server public key alias
-keystore server-keystore.jks -dname "cn=server" -keypass Private server key
password -storepass KeyStore password
```

3. To verify that the key has been correctly created, run the following keytool command:

```
keytool -list -keystore server-keystore.jks -keypass Server private key
password -storepass KeyStore password
```

4. To use the key, sign it by running the following keytool command:

```
keytool -selfcert -validity 5000 -alias Server public key alias -keystore
server-keystore.jks
-keypass Private server key password -storepass KeyStore password
```

5. Export the server public key to the server keystore by running the following keytool command:

```
keytool -export -alias Server public key alias -keystore server-keystore.jks
-file server.pubkey -keypass Server private key password -storepass KeyStore
password
```

Verifying Signatures of Trusted Clients

To verify signatures of trusted clients, import the client public key into the server keystore:

1. In your development environment, go to *JDEV_HOME/jdk/bin* and open the command prompt.

2. To verify the signature of trusted clients, import the client's public key in to the server keystore by running the following keytool command:

```
keytool -import -alias Client public key alias -file client.pubkey -keystore
server-keystore.jks -keypass Private server key password -storepass KeyStore
password
```

3. Import the server public key into the client keystore by running the following keytool command:

```
keytool -import -alias Server public key alias -file server.pubkey -keystore
client-keystore.jks -keypass Private key password -storepass KeyStore password
```

When the tool prompts you if the key is self certified, you must enter Yes.

[Example 9–1](#) shows a sample output that is generated after this procedure is completed successfully.

Example 9–1 Sample Output Generated by the Keytool

```
[user@server]$ keytool -import -alias client -file client.pubkey
-keystore server-keystore.jks -keypass Server private key password -storepass
Keystore password
Owner: CN=client
Issuer: CN=client
```

Serial number: *serial number*, for example, 123a19cb
 Valid from: *Date, Year, and Time* until: *Date, Year, and Time*
 Certificate fingerprints:
 ...
 Trust this certificate? [no]: *yes*
 Certificate was added to keystore.

Securing Identity Propagation

To secure identity propagation, you must configure SSL on Oracle Content Server.

1. Log into Oracle Content Server as an administrator.
2. From Administration, choose **Providers**. The Create a New Provider page displays.
3. Click **Add** for **sslincoming**. The Add Incoming Provider page displays.
4. In **Provider Name**, enter a name for the provider, for example, `sslincomingprovider`.

When the new provider is set up, a directory with the provider name is created as a subdirectory of the `CONTENT_SERVER_HOME/data/providers` directory.

5. In **Provider Description**, briefly describe the provider, for example, `SSL Incoming Provider for securing the Content Server`.
6. In **Provider Class**, enter the class of the `sslincoming` provider, for example, `idc.provider.ssl.SSLSocketIncomingProvider`.

Note: You can add a new SSL keepalive incoming socket provider or a new SSL incoming socket provider. Using a keepalive socket improves the performance of a session and is recommended for most implementations.

7. In **Connection Class**, enter the class of the connection, for example, `idc.provider.KeepaliveSocketIncomingConnection`.
8. In **Server Thread Class**, enter the class of the server thread, for example, `idc.server.KeepaliveIdcServerThread`.
9. In **Server Port**, specify an open server port, for example, 5555.
10. Select the **Require Client Authentication** checkbox.
11. In **Keystore password**, enter the password to access the keystore.
12. In **Alias**, enter the alias of the keystore.
13. In **Alias password**, enter the password of the alias.
14. In **Truststore password**, enter the password of the trust store.
15. Click **Add**. The new incoming provider is now set up.
16. Go to the new provider directory that was created in step 4.
17. To specify truststore and keystore, create a file named `sslconfig.hda`.
18. Copy the server keystore to the server.
19. Configure the `sslconfig.hda` file. [Example 9-2](#) shows how the `.hda` file should look after you include the truststore and keystore information.

Example 9–2 Sample `sslconfig.hda` File

```
@Properties LocalData
TruststoreFile=/tmp/ssl/server_keystore
KeystoreFile=/tmp/ssl/server_keystore
@end
```

9.7.2 Configuring the Load Balancer

You must configure the load balancer to act as a socket load balancer. This will be used to make socket connections from the Oracle HTTP Server as well as WebCenter.

1. Configure load balancer addresses.

A virtual address should be configured on the load balancer. This address will only be used internally to access Oracle Content Server (OCS). This address should be configured to route to the OCS cluster.

Example:

Virtual host on load balancer: `wcinternal.mycompany.com:9054`

Maps to: `WCHOST1:9054, WCHOST2:9054`

Load-balancing method: `round-robin`

2. Reconfigure Oracle HTTP Server and Content Server.

Oracle HTTP Server and Content Server both must be reconfigured to support the new virtual address.

- a. On the Content Server machine (WCHOST1), in the file `<content_server_dir>/config/config.cfg`, replace the two occurrences of `WCHOST1` (under `@Properties LocalData` and `@ResultSet ProxiedServers`) with the virtual host name `wcinternal.mycompany.com`.

Ensure that the LBR IP address is added to "SocketHostAddressSecurityFilter".

- b. **Restart the Content Server server.**
 - c. Copy the `SecurityInfo.hda` file to the same directory on the Oracle HTTP Server host. This should overwrite the copy of the file already there.
3. Restart Oracle HTTP Server. Verify that the address and login work as before.

9.7.3 Installing Content Server Cluster

Each Content Server node is installed using the cluster installer which can be found in `<content_server_dir>/bin`.

Before proceeding, shut down the Content Server and Administration Server from the previous installation. The installation process consists of the following steps:

- [Section 9.7.3.1, "Installing the First Content Server Node"](#)
- [Section 9.7.3.2, "Installing the Second Content Server Node"](#)
- [Section 9.7.3.3, "Post-Installation Steps"](#)
- [Section 9.7.3.4, "Starting the Servers"](#)

9.7.3.1 Installing the First Content Server Node

From the first node (WCHOST1), run the following command (all on one line):

```
Installer
```

```

-set-ClusterNodeIntradocDir=<ocs-stub>
-set-ClusterNodeName=<node_name>
-set-ClusterNodeAddress=<ip_of_node>
-set-ClusterBinDirRule=<local|shared>
ConfigureClusterNode
ConfigureAdminClusterNode

```

For <node_name>, specify WCHOST1.

For <ocs-stub>, use a local directory, for example, /u01/app/oracle/product/ucm.

For Set-ClusterBinDirRule, specify local.

9.7.3.2 Installing the Second Content Server Node

From the second node (WCHOST2), run the command (all on one line):

```

Installer
-set-ClusterNodeIntradocDir=<ocs-stub>
-set-ClusterNodeName=<node_name>
-set-ClusterNodeAddress=<ip_of_node>
-set-ClusterBinDirRule=<local|shared>
ConfigureClusterNode
ConfigureAdminClusterNode

```

For <node_name>, specify WCHOST2.

For <ocs-stub>, use a local directory, for example, /u01/app/oracle/product/ucm.

For Set-ClusterBinDirRule, specify local.

9.7.3.3 Post-Installation Steps

After setting up the nodes, add the following lines to the <ocs-stub>/bin/intradoc.cfg and the <ocs-stub>/admin/bin/intradoc.cfg files on each of the nodes WCHOST1 and WCHOST2:

```

DisableSharedCacheChecking=true
ClusterNodeName=<node_name>
ClusterGroup=<cluster_name>
SocketServerAddress=<server_IP_address>
UserPublishCacheDir=<ocs-stub>/data/users
SharedDir=<ocs-stub>/shared
ArchiverDoLocks=true

```

<node_name> is the identifier for the node.

<cluster_name> is the identifier for the cluster.

All nodes should have a different node name, but should have the same cluster group name. For example, WCHOST1 would be configured as follows:

```

DisableSharedCacheChecking=true
ClusterNodeName=WCHOST1
ClusterGroup=ucmcluster
SocketServerAddress=<IP_of_WCHOST1>
UserPublishCacheDir=<Stub_Location_on_WCHOST1>/data/users
SharedDir=<Stub_Location_on_WCHOST1>/shared
ArchiverDoLocks=true

```

And WCHOST2 would be configured as follows:

```

DisableSharedCacheChecking=true

```

```
ClusterNodeName=WCHOST2
ClusterGroup=ucmcluster
SocketServerAddress=<IP_of_WCHOST2>
UserPublishCacheDir=<Stub_Location_on_WCHOST2>/data/users
SharedDir=<Stub_Location_on_WCHOST2>/shared
ArchiverDoLocks=true
```

9.7.3.4 Starting the Servers

Start Oracle Content Server and Administration Server on each node using the binaries in `<ocs-stub>/bin`.

Verify that the log and PID files are created in the `<ocs-stub>/admin/etc` and `<ocs-stub>/etc` directories.

Start Oracle HTTP Server on WEBHOST1.

9.8 Adding Another Oracle HTTP Server and Configuring a Load Balancer

This section contains the following sections:

- [Section 9.8.1, "Configuring the Second Oracle HTTP Server"](#)
- [Section 9.8.2, "Configuring the Load Balancer"](#)
- [Section 9.8.3, "Configuring the Content Server's HTTP Address"](#)

9.8.1 Configuring the Second Oracle HTTP Server

To configure the second Oracle HTTP Server (that is, the Oracle HTTP Server on WEBHOST2), perform the steps in [Section 9.4.2, "Steps to Perform on the Oracle HTTP Server Machine."](#)

Validate that the Content Server is accessible at `http://WEBHOST2:7777/idc`.

9.8.2 Configuring the Load Balancer

The load balancer in the WebCenter enterprise topology should already be able to send requests from the load balancer address:

```
http://wcinternal.mycompany.com/idc
```

to both

```
http://webhost1:7777/idc
```

and

```
http://webhost2:777/idc
```

Validate the above before continuing.

9.8.3 Configuring the Content Server's HTTP Address

The Content Server's HTTP address must be reconfigured to point from WEBHOST1 to the load balancer.

In the file `<content_server_dir>/config/config.cfg`, edit the `HttpServerAddress` line to:

```
HttpServerAddress=wcinternal.mycompany.com:80
```

9.9 Reconfiguring the Content Server to Support RAC

In the file `<content_server_dir>/config/config.cfg`, edit the `JdbcConnectionString` line to provide an RAC connection string. For example:

```
JdbcConnectionString=jdbc:oracle:thin:@(DESCRIPTION=
  (ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP) (HOST=RACHOST1) (PORT=1521))
    (ADDRESS=(PROTOCOL=TCP) (HOST=RACHOST2) (PORT=1521))
  ) (CONNECT_DATA=(SERVICE_NAME=wcedg)))
```

9.10 Backing Up the Installation

Back up all the installation files on both the Web machine and the Content Server machine.

For the HTTP Server machine, back up the configuration directory:

```
WEBHOST1> tar -cvpf ucmWHconfigback.tar /shared/oracle/ucm/server
```

For the Content Server machine, back up all install files and configuration:

```
WCHOST1> tar -cvpf ucmCSback.tar /shared/oracle/ucm/server
```

In addition, back up the cluster configuration files:

```
WCHOST1> tar -cvpf ucmCS1back.tar <ocs-stub>
WCHOST2> tar -cvpf ucmCS2back.tar <ocs-stub>
```

Integration With Oracle Identity Management

This chapter describes how to integrate Oracle WebCenter with Oracle Identity Management. It contains the following sections:

- [Section 10.1, "Credential and Policy Store Configuration"](#)
- [Section 10.2, "Oracle Access Manager Integration"](#)
- [Section 10.3, "Configuring WebCenter Applications"](#)
- [Section 10.4, "Configuring WebCenter and BPEL Authentication"](#)
- [Section 10.5, "Backing Up the Installation"](#)

10.1 Credential and Policy Store Configuration

The following topics describe credential and policy store configuration in detail:

- [Section 10.1.1, "Overview of Credential and Policy Store Configuration"](#)
- [Section 10.1.2, "Credential Store Configuration"](#)
- [Section 10.1.3, "Policy Store Configuration"](#)
- [Section 10.1.4, "Reassociation of Credentials and Policies"](#)

10.1.1 Overview of Credential and Policy Store Configuration

Oracle Fusion Middleware allows using different types of credential and policy stores in a WebLogic domain. Domains can use stores based on an XML file or on different types of LDAP providers. When a domain uses an LDAP store, all policy and credential data is kept and maintained in a centralized store. However, when using XML policy stores, the changes made on Managed Servers are not propagated to the Administration Server unless they use the same domain home. The Oracle Fusion Middleware WebCenter EDG topology uses different domain homes for the Administration Server and the Managed Server, thus Oracle requires the use of an LDAP store as policy and credential store for integrity and consistency. By default Oracle WebLogic Server domains use an XML file for the policy store. The following sections describe the steps required to change the default store to Oracle Internet Directory LDAP for credentials or policies.

Note: The backend repository for the policy store and the credential store must use the same kind of LDAP server. To preserve this coherence, note that reassociating one store implies reassociating the other one, that is, the reassociation of both the credential and the policy stores is accomplished as a unit using the Fusion Middleware Control or the WLST command `reassociateSecurityStore`. For more information, see [Section 10.1.4, "Reassociation of Credentials and Policies."](#)

10.1.2 Credential Store Configuration

A credential store is a repository of security data (credentials). A credential can hold user name and password combinations, tickets, or public key certificates. Credentials are used during authentication, when principals are populated in subjects, and, further, during authorization, when determining what actions the subject can perform. In this section, steps are provided to configure Oracle Internet Directory LDAP as a credential store for the Oracle Fusion Middleware WebCenter EDG topology. For more details on credential store configuration, refer to the "Configuring the Credential Store" chapter in the *Oracle Fusion Middleware Security Guide*.

The following section describe credential store configuration:

- [Section 10.1.2.1, "Creating the LDAP Authenticator"](#)
- [Section 10.1.2.2, "Moving the WebLogic Administrator to LDAP"](#)
- [Section 10.1.2.3, "Reassociating the Domain Credential Store"](#)

10.1.2.1 Creating the LDAP Authenticator

To be safe, before you create the LDAP authenticator, you should first back up the relevant configuration files:

```
ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>/config/config.xml
ORACLE_BASE/admin/<domain_name>/aserver/<domain_
name>/config/fmwconfig/jps-config.xml
ORACLE_BASE/admin/<domain_name>/aserver/<domain_
name>/config/fmwconfig/system-jazn-data.xml
```

Also back up the `boot.properties` file for the Administration Server.

To configure the credential store to use LDAP, set the proper authenticator using the WebLogic Server Console:

1. Log in to the WebLogic Server Console.
2. Click the **Security Realms** link on the left navigational bar.
3. Click the **myrealm** default realm entry to configure it.
4. Open the **Providers** tab within the realm.
5. Observe that there is a `DefaultAuthenticator` provider configured for the realm.
6. Click the **New** button to add a new provider.
7. Enter a name for the provider such as **OIDAuthenticator** or **OVDAuthenticator** depending on whether Oracle Internet Directory or Oracle Virtual Directory will be used.

8. Select the **OracleInternetDirectoryAuthenticator** or **OracleVirtualDirectoryAuthenticator** type from the list of authenticators depending on whether Oracle Internet Directory or Oracle Virtual Directory will be used.
9. Click **OK**.
10. In the Providers screen, click the newly created Authenticator.
11. Set the control flag to **SUFFICIENT**. This indicates that if a user can be authenticated successfully by this authenticator, then it should accept that authentication and should not continue to invoke any additional authenticators. If the authentication fails, it will fall through to the next authenticator in the chain. Make sure all subsequent authenticators also have their control flag set to **SUFFICIENT**; in particular, check the DefaultAuthenticator and set that to **SUFFICIENT**.
12. Click **Save** to save this setting.
13. Open the **Provider Specific** tab to enter the details for the LDAP server.
14. Enter the details specific to your LDAP server, as shown in the following table:

Parameter	Value	Value Description
Host	For example: oid.mycompany.com	The LDAP server's server ID.
Port	For example: 636	The LDAP server's port number.
Principal	For example: cn=orcladmin	The LDAP user DN used to connect to the LDAP server.
Credential	NA	The password used to connect to the LDAP server
SSL Enabled	Checked	Specifies whether SSL protocol is used when connecting to LDAP server.
User Base DN	For example: cn=users,dc=us,dc=mycompany,dc=com	Specify the DN under which your Users start.
Group Base DN	For example: cn=groups,dc=us,dc=mycompany,dc=com	Specify the DN that points to your Groups node.
Use Retrieved User Name as Principal	Checked	Must be turned on.

Click **Save** when done.

15. Click **Activate Changes** to propagate the changes.

10.1.2.1.1 Adding the Jive Administrator Before moving the WebLogic Administrator from defaultProvider to OID LDAP or External LDAP, follow these steps:

1. Identify which user on OID LDAP or external LDAP server (fmwadmin) will be assigned administrator privileges in Jive.
2. Once you have identified the user, go to the Weblogic Console and add this user to the security default provider.
3. Go to Jive Administration Console and log in as the WebLogic admin user in emdedded LDAP, and add the user (fmwadmin) as the administrator for Jive.

10.1.2.1.2 Setting the Order of Providers Reorder the OID/OVD Authenticator and Default Authenticator and ensure that the control flag for each authenticator is set as follows:

- OID LDAP Authenticator: **SUFFICIENT**
- Default Authenticator: **SUFFICIENT**

Restart the Administration Server so that the changes are effective, as follows:

1. Stop the Administration Server:

```
SOAHOST1> cd ORACLE_BASE/admin/domainName/aserver/domainName/bin
SOAHOST1> ./stopWebLogic.sh
```

2. Start the Administration Server:

```
SOAHOST1> ./startWebLogic.sh
```

Note: The reassociation of both the credential and the policy stores is accomplished as a unit using Fusion Middleware Control or the WLST command `reassociateSecurityStore`.

For more details on credential store configuration, refer to the "Configuring the Credential Store" chapter in the *Oracle Fusion Middleware Security Guide*.

10.1.2.2 Moving the WebLogic Administrator to LDAP

This section provides details for provisioning a new administrator user and group for managing the Oracle Fusion Middleware WebCenter WebLogic Domain. This section describes the following tasks:

- [Section 10.1.2.2.1, "Provisioning Admin Users and Groups in an LDAP Directory"](#)
- [Section 10.1.2.2.2, "Assigning the Admin Role to the Admin Group"](#)
- [Section 10.1.2.2.3, "Updating the boot.properties File and Restarting the System"](#)

10.1.2.2.1 Provisioning Admin Users and Groups in an LDAP Directory As mentioned in the introduction to this section, users and groups from multiple WebLogic domains may be provisioned in a central LDAP user store. In such a case, there is a possibility that one WebLogic admin user may have access to all the domains within an enterprise. This is not a desirable situation. To avoid this, the users and groups provisioned must have a unique distinguished name within the directory tree. In this guide, the admin user and group for the WebCenter EDG WebLogic domain will be provisioned with the DNs below:

- Admin User DN:

```
cn=weblogic_wc,cn=Users,dc=us,dc=mycompany,dc=com
```

- Admin Group DN:

```
cn=WC_Administrators,cn=Groups,dc=us,dc=mycompany,dc=com
```

Follow these steps to provision the admin user and admin group in Oracle Internet Directory:

1. Create an ldif file named `admin_user.ldif` with the contents shown below and then save the file:

```

dn: cn=weblogic_wc, cn=Users, dc=us, dc=mycompany, dc=com
orclsamaccountname: weblogic_wc
givenname: weblogic_wc
sn: weblogic_wc
userpassword: Welcome1
obver: 10.1.4.0
mail: weblogic_wc
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetorgperson
objectclass: orcluser
objectclass: orcluserV2
objectclass: oblixorgperson
uid: weblogic_wc
cn: weblogic_wc
description: Admin User for the IDM Domain

```

2. Run the `ldapadd` command located under the `ORACLE_HOME/bin` directory to provision the user in Oracle Internet Directory.

Note: The `ORACLE_HOME` used here is the `ORACLE_HOME` for the Identity Management installation where Oracle Internet Directory resides.

For example (the command is shown as two lines in the example below for readability purposes, but you should enter the command on a single line):

```

OIDHOST1> ORACLE_HOME/bin/ldapadd -h oid.mycompany.com -p 389 -D
cn="orcladmin" -w welcome1 -c -v -f admin_user.ldif

```

3. Create an ldif file named `admin_group.ldif` with the contents shown below and then save the file:

```

dn: cn=WC_Administrators, cn=Groups, dc=us, dc=mycompany, dc=com
displayname: WC_Administrators
objectclass: top
objectclass: groupOfUniqueNames
objectclass: orclGroup
uniquemember: cn=weblogic_wc,cn=users,dc=us,dc=mycompany,dc=com
cn: WC_Administrators
description: Administrators Group for the SOA Domain

```

4. Run the `ldapadd` command located under the `ORACLE_HOME/bin/` directory to provision the group in Oracle Internet Directory (the command is shown as two lines in the example below for readability purposes, but you should enter the command on a single line):

```

OIDHOST1> ORACLE_HOME/bin/ldapadd -h oid.mycompany.com -p 389 -D
cn="orcladmin" -w welcome1 -c -v -f admin_group.ldif

```

10.1.2.2.2 Assigning the Admin Role to the Admin Group After adding the users and groups to Oracle Internet Directory, the group must be assigned the Admin role within the WebLogic domain security realm. This enables all users that belong to the group to be administrators for that domain. Follow these steps to assign the Admin role to the Admin group:

1. Log into the WebLogic Administration Server Console.

2. In the left pane of the console, click **Security Realms**.
3. On the Summary of Security Realms page, click **myrealm** under the Realms table.
4. On the Settings page for myrealm, click the Roles & Policies tab.
5. On the Realm Roles page, expand the Global Roles entry under the Roles table. This brings up the entry for Roles. Click on the **Roles** link to bring up the Global Roles page.
6. On the Global Roles page, click the Admin Role to bring up the Edit Global Role page:
 - a. On the Edit Global Roles page, under the Role Conditions table, click the **Add Conditions** button.
 - b. On the Choose a Predicate page, select **Group** from the drop down list for predicates and click **Next**.
 - c. On the Edit Arguments Page, specify **WC_Administrators** in the **Group Argument** field and click **Add**.
7. Click **Finish** to return to the Edit Global Rule page.
8. The Role Conditions table now shows the WC_Administrators Group as an entry.
9. Click **Save** to finish adding the Admin Role to the WC_Administrators Group.
10. Validate that the changes were successful by bringing up the WebLogic Administration Server Console using a web browser. Log in using the credentials for the weblogic_wc user.

10.1.2.2.3 Updating the boot.properties File and Restarting the System The `boot.properties` file for the Administration Server should be updated with the WebLogic admin user created in Oracle Internet Directory. Follow the steps below to update the `boot.properties` file:

1. On SOAHOST1, go the following directory:

```
SOAHOST1>cd ORACLE_BASE/admin/domainName/aserver/domainName/servers/  
AdminServer/security
```
2. Rename the existing `boot.properties` file:

```
SOAHOST1> mv boot.properties boot.properties.backup
```
3. Use a text editor to create a file called `boot.properties` under the security directory. Enter the following lines in the file:

```
username=adminUser  
password=adminUserPassword
```
4. Save the file.
5. Stop the Administration Server:

```
SOAHOST1> cd ORACLE_BASE/admin/domainName/aserver/domainName/bin  
SOAHOST1> ./stopWebLogic.sh
```
6. Start the Administrator Server:

```
SOAHOST1> ./startWebLogic.sh
```

10.1.2.3 Reassociating the Domain Credential Store

The reassociation of both the Credential and the Policy stores is accomplished as a unit using Fusion Middleware Control or the WLST command `reassociateSecurityStore`. See [Section 10.1.4, "Reassociation of Credentials and Policies"](#) for detailed steps

10.1.3 Policy Store Configuration

The domain policy store is the repository of system and application-specific policies. In a given domain, there is one store that stores all policies that all applications deployed in the domain may use. This section provides the steps to configure Oracle Internet Directory LDAP as the policy store for the Oracle Fusion Middleware WebCenter EDG topology. For more details on policy store configuration, refer to the "OPSS Authorization and the Policy Store" chapter in the Oracle Fusion Middleware Security Guide. *Oracle Fusion Middleware Security Guide*.

10.1.3.1 Prerequisites to Using an LDAP-Based Policy Store

In order to ensure the proper access to an LDAP server directory (Oracle Internet Directory) used as a policy store, you must set a node in the server directory.

An Oracle Internet Directory administrator must follow these steps to create the appropriate node in an Oracle Internet Directory Server:

1. Create an LDIF file (assumed to be `jpstestnode.ldif` in this example) specifying the following DN and CN entries:

```
dn: cn=jpsroot_wc
cn: jpsroot_wc
objectclass: top
objectclass: OrclContainer
```

The distinguished name of the root node (illustrated by the string `jpsroot_wc` above) must be distinct from any other distinguished name. One root node can be shared by multiple WebLogic domains. It is not required that this node be created at the top level, as long as read and write access to the subtree is granted to the Oracle Internet Directory administrator.

2. Import this data into Oracle Internet Directory server using the command `ldapadd`, as illustrated in the following example (the command is shown as two lines in the example below for readability purposes, but you should enter the command on a single line):

```
OIDHOST1> ORACLE_HOME/bin/ldapadd -h ldap_host -p ldap_port -D
cn=orcladmin -w password -c -v -f jpstestnode.ldif
```

3. Verify that the node has been successfully inserted using the command `ldapsearch`, as illustrated in the following example (the command is shown as two lines in the example below for readability purposes, but you should enter the command on a single line):

```
OIDHOST1> ORACLE_HOME/bin/ldapsearch -h ldap_host -p ldap_port -D
cn=orcladmin -w password -b "cn=jpsroot_wc" objectclass="orclContainer"
```

4. When using Oracle internet Directory as the LDAP-Based Policy Store run the utility `oidstats.sql` in the INFRADBHOSTs to generate database statistics for optimal database performance:

```
OIDHOSTn> ORACLE_HOME/ldap/admin/oidstats.sql
```

The `oidstats.sql` utility must be run just once after the initial provisioning. For details about this utility, consult the *Oracle Fusion Middleware User Reference for Oracle Identity Management*.

10.1.3.2 Cataloging Oracle Internet Directory Attributes

An Oracle Internet Directory attribute used in a search filter must be indexed. The command `ldapmodify`, whose syntax is illustrated below, can also be used to index attributes specified in an LDIF file (the command is shown as two lines in the example below for readability purposes, but you should enter the command on a single line):

```
OIDHOST1> ORACLE_HOME/bin/ldapmodify -h host -p port -D bindDN -w  
bindPassword -v -f catalogue-modify-ldif-filename
```

For example, the above command can be used with the following sample LDIF file to catalog the attributes `createtimestamp` and `modifytimestamp`:

```
dn: cn=catalogs  
changetype: modify  
add: orclindexedattribute  
orclindexedattribute: modifytimestamp  
orclindexedattribute: createtimestamp
```

Each of the following Oracle Internet Directory attributes must be indexed:

```
orclrolescope  
orclassignedroles  
orclApplicationCommonName  
orclAppFullName  
orclCSFAlias  
orclCSFKey  
orclCSFName  
orclCSFDBUrl  
orclCSFDBPort  
orclCSFCredentialType  
orclCSFExpiryTime  
modifytimestamp  
createtimestamp  
orcljpsassignee
```

10.1.3.3 Reassociating the Domain Policy Store

Reassociating the policy store consists in migrating policy data from a file- or LDAP-based repository to an LDAP-based repository, that is, reassociation changes the repository preserving the integrity of the data stored. For each policy in the source policy store, reassociation searches the target LDAP directory and, if it finds a match, it updates the matching policy as appropriate. If none is found, it simply migrates the policy as is.

At any time, after a domain policy store has been instantiated, a file- or LDAP-based policy store can be reassociated into an LDAP-based policy store storing the same data. To support it, the domain has to be configured, as appropriate, to use an LDAP policy store.

The reassociation of both the credential and the policy stores is accomplished as a unit using Fusion Middleware Control or the WLST command `reassociateSecurityStore`. See [Section 10.1.4, "Reassociation of Credentials and Policies"](#) for detailed steps.

10.1.4 Reassociation of Credentials and Policies

To reassociate the policy and credential store with Oracle Internet Directory, use the WLST `reassociateSecurityStore` command. Follow these steps:

1. From SOAHOST1, start the `wlst` shell:

```
SOAHOST1>cd ORACLE_COMMONHOME/common/bin
SOAHOST1>./wlst.sh
```

2. Connect to the WebLogic Administration Server using the `wlst connect` command shown below:

Syntax:

```
connect('AdminUser', "AdminUserPassword", t3://hostname:port)
```

For example:

```
connect("weblogic", "welcome1", "t3://SOAHOST1VHN1:7001")
```

3. Run the `reassociateSecurityStore` command as shown below:

Syntax:

```
reassociateSecurityStore(domain="domainName", admin="cn=orcladmin",
password="orclPassword", ldapurl="ldap://LDAPHOST:LDAPPOR", servertype="OID",
jpsroot_wc="cn=jpsroot_wc")
```

For example:

```
wls:/SOAEDGDomain/serverConfig>reassociateSecurityStore(domain="soaedg_domain",
admin="cn=orcladmin",password="welcome1",ldapurl="ldap://oid.mycompany.com:389"
,servertype="OID",jpsroot_wc="cn=jpsroot_wc")
```

The output for the command is shown below:

```
{servertype=OID,jpsroot_wc=cn=jpsroot_wc_idm_idmhost1,admin=cn=orcladmin,
domain=IDMDomain,ldapurl=ldap://oid.mycompany.com:389,password=welcome1}
Location changed to domainRuntime tree. This is a read-only tree with
DomainMBean as the root.
```

For more help, use `help(domainRuntime)`

```
Starting Policy Store reassociation.
LDAP server and ServiceConfigurator setup done.
```

```
Schema is seeded into LDAP server
Data is migrated to LDAP server
Service in LDAP server after migration has been tested to be available
Update of jps configuration is done
Policy Store reassociation done.
Starting credential Store reassociation
LDAP server and ServiceConfigurator setup done.
Schema is seeded into LDAP server
Data is migrated to LDAP server
Service in LDAP server after migration has been tested to be available
Update of jps configuration is done
Credential Store reassociation done
Jps Configuration has been changed. Please restart the server.
```

4. Restart the Administration Server after the command completes successfully.

Note: For credential and policy changes to take effect, the servers in the domain must be restarted.

10.2 Oracle Access Manager Integration

This section describes how to set up Oracle Access Manager as the single sign-on solution for the Oracle WebCenter EDG topology.

10.2.1 Overview of Oracle Access Manager Integration

Oracle Access Manager (OAM) is the recommended single sign-on solution for Oracle Fusion Middleware 11g Release 1. For more information on installing and configuring an OAM installation, see *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*. This chapter explains the procedure for configuring the WebCenter installation with an existing OAM installation and the underlying directory service. Oracle recommends using either Oracle Internet Directory (OID) or Oracle Virtual Directory (OVD) or both of these directory services.

Note: The WebCenter EDG topology described in this book uses a Single Sign-On configuration where both the WebCenter System and the Single Sign-On System are in the same network domain (mycompany.com) For a multi-domain configuration, please refer to the required configuration steps in "Chapter 7, Configuring Single Sign-On," of the *Oracle Access Manager Access Administration Guide*.

10.2.2 Prerequisites for Oracle Access Manager

The setup for Oracle Access Manager (OAM) assumes an existing OAM installation complete with Access Managers and a policy protecting the Policy Manager. For more information on installing and configuring an OAM installation, see *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*. This setup includes a directory service such as Oracle Internet Directory (OID) either as a stand-alone or as part of an Oracle Virtual Directory (OVD) configuration. This chapter will provide the necessary steps for configuring your WebCenter installation with either OID or OVD.

In addition, the OAM installation should have its own Web server configured with WebGate. This section also provides the steps for using the OAM Web server as a delegated authentication server.

10.2.3 Using the OAM Configuration Tool

The OAM Configuration Tool (oamcfg) starts a series of scripts and setup the required policies. It requires various parameters as inputs. Specifically, it creates the following:

1. A Form Authentication scheme in OAM
2. Policies to enable authentication in WebLogic Server
3. A WebGate entry in OAM to enable Oracle HTTP Server WebGates (from your Web Tier) to protect your configured application
4. A Host Identifier, depending on the scenario chosen (a default host identifier would be used, if not provided)

5. Policies to protect and unprotect application specific URLs.

This section covers the following topics:

- [Section 10.2.3.1, "Collecting the Information for the OAM Configuration Tool"](#)
- [Section 10.2.3.2, "Running the OAM Configuration Tool"](#)
- [Section 10.2.3.3, "Verifying Successful Creation of the Policy Domain and AccessGate"](#)
- [Section 10.2.3.4, "Updating the Host Identifier"](#)
- [Section 10.2.3.5, "Updating the WebGate Profile"](#)
- [Section 10.2.3.6, "Adding Additional Access Servers"](#)
- [Section 10.2.3.7, "Configure Delegated Form Authentication"](#)

10.2.3.1 Collecting the Information for the OAM Configuration Tool

The following information should be collected or prepared prior to running the OAM Configuration tool:

1. **Password:** Create a secure password. This will be used as the password for the WebGate installation created later.
2. **LDAP Host:** host name of the Directory Server or Load Balancer address in the case of an HA/EDG configuration.
3. **LDAP Port:** port of the Directory Server.
4. **LDAP USER DN:** DN of the LDAP admin user. This will be a value such as "cn=orcladmin."
5. **LDAP password:** password of the LDAP admin user
6. **oam_aa_host:** host name of an Oracle Access Manager
7. **oam_aa_port:** port of the Oracle Access Manager

10.2.3.2 Running the OAM Configuration Tool

The OAM Configuration Tool resides in the `ORACLE_HOME/modules/oracle.oamprovider_11.1.1/directory` (`ORACLE_HOME` will depend on which machine you are running this). The tool can be run from any machine with the required installation files. In this case, we run it from `SOAHOST1`.

The OAM Configuration Tool should be run as follows (all on a single command line):

```
MW_HOME/jrockit_160_14_R27.6.4-18/bin java -jar oamcfgtool.jar mode=CREATE
app_domain="WebCenter_EDG"
protected_uris="$URI_LIST"
public_uris="$PUBLIC_URI_LIST"
app_agent_password=<Password_to_be_provisioned_for_App_Agent>
ldap_host=OID.MYCOMPANY.COM
ldap_port=389
ldap_userdn="cn=orcladmin"
ldap_userpassword=<Password_of_LDAP_Admin_User>
oam_aaa_host=OAMHOST1
oam_aaa_port=OAMPOR1
```

The `$URI_LIST` and `$PUBLIC_URI_LIST` variables in the above command depend on the topology:

- **WebCenter only:**

```
$URI_LIST="/webcenter/adfAuthentication,/owc_wiki/user/login.jz,/owc_wiki/adfAuthentication,/integration/worklistapp,/workflow/sdpmessagingsca-ui-worklist/faces/adf.task-flow,/workflow/WebCenterWorklistDetail/faces/adf.task-flow,/workflow/sdpmessagingsca-ui-worklist,/rss/rsservlet,/owc_discussions/login!withRedirect.jspa,/owc_discussions/login!default.jspa,/owc_discussions/login.jspa,/owc_discussions/admin,/em,/console"
```

```
$PUBLIC_URI_LIST="/webcenter,/owc_wiki,/owc_discussions,/rss,/workflow"
```

- **WebCenter and SOA:**

```
$URI_LIST="/webcenter/adfAuthentication,/owc_wiki/user/login.jz,/owc_wiki/adfAuthentication,/integration/worklistapp,/workflow/sdpmessagingsca-ui-worklist/faces/adf.task-flow,/workflow/WebCenterWorklistDetail/faces/adf.task-flow,/workflow/sdpmessagingsca-ui-worklist,/rss/rsservlet,/owc_discussions/login!withRedirect.jspa,/owc_discussions/login!default.jspa,/owc_discussions/login.jspa,/owc_discussions/admin,/em,/console,/DefaultToDoTaskFlow,/b2b,/sdpmessaging/userprefs-ui"
```

```
$PUBLIC_URI_LIST="/webcenter,/owc_wiki,/owc_discussions,/rss,/workflow"
```

Note: If SOA is installed later or other additional URLs need to be protected, the OAM configuration tool should be executed again using the same `app_domain` and including *all* the URLs that would be protected (not just the new ones).

If your command ran successfully, you should see the following output:

```
Successfully connected to LDAP
Processed input parameters
Intialized Global Configuration
```

10.2.3.3 Verifying Successful Creation of the Policy Domain and AccessGate

Verifying the Policy Domain

To verify the policy domain, complete these steps:

1. Log on to the Oracle Access Manager:

```
http://OAMADMINHOST:<port>/access/oblix/
```
2. Click **Policy Manager**.
3. Click the **My Policy Domains** link on the left panel, you will see a list of all policy domains, among which the domain you just created will be listed. It will have the suffix `_PD` (for example, `WebCenter_EDG_PD`). In the third column (URL prefixes), you will also see the URIs you specified during the creation of this domain).
4. Click the link to the policy domain you just created. you will land in the General area of this domain.
5. Click the **Resources** tab, you will see the URIs you specified. You can also click other tabs to view other settings.

Verifying the AccessGate Configuration

To verify the AccessGate configuration, complete these steps:

1. Click the **Access System Console** link on the top right hand side (this acts like a toggle; after you click it, it becomes the **Policy Manager** link).
2. Click the **Access System Configuration** tab.
3. Click the **AccessGate Configuration** link on the left panel.
4. Enter 'WebCenter_EDG' as the search criterion (or any other substring you may have used as the `app_domain` name in [Section 10.2.3.2, "Running the OAM Configuration Tool"](#)), and click **Go**.
5. Once the AccessGate for the domain you just created shows up (this will have the suffix `_AG` (for example, **WebCenter_EDG_AG**), click it, and the details of the AccessGate you just created appear.

10.2.3.4 Updating the Host Identifier

The OAM Configuration Tool uses the value of the `app_domain` parameter to create a host identifier for the policy domain. This host identifier must be updated with all the host name variations for the host so that the configuration works correctly. Follow the steps below to update the host identifier created by the OAM Configuration Tool:

1. Navigate to the Access System Console by specifying the following URL in your web browser:

```
http://hostname:port/access/oblix
```

where *hostname* refers to the host where WebPass Oracle HTTP Server instance is running and *port* refers to the HTTP port of the Oracle HTTP Server instance.

2. When prompted for a username and password, log in as an administrator. Click **OK**.
3. On the Access System main page, click the **Access System Console** link.
4. On the Access System Console page, click the Access System Configuration tab.
5. On the Access System Configuration page, click **Host Identifiers** at the bottom left.
6. On the List all host identifiers page, click on the host identifier created by the OAM Configuration Tool. For example, select `WebCenter_EDG`.
7. On the Host Identifier Details page, click **Modify**.
8. On the Modifying host identifier page, add all the possible host name variations for the host. Click the plus and minus symbols to add or delete fields as necessary. The **Preferred HTTP Host** value used in the Access System Configuration must be added as one of the host name variations. For example: `wcedg_wd`, `webhost1.mycompany.com:7777`, `admin.mycompany.com:7777`.
9. Select the check box next to Update Cache and then click **Save**.
A message box with the following message is displayed: "Updating the cache at this point will flush all the caches in the system. Are you sure?".
Click **OK** to finish saving the configuration changes.
10. Verify the changes on the Host Identifier Details page.

10.2.3.5 Updating the WebGate Profile

The OAM Configuration Tool populates the `Preferred_HTTP_Host` and `hostname` attributes for the WebGate profile that is created with the value of the `app_domain` parameter. Both these attributes must be updated with the proper values for the

configuration to work correctly. Follow the steps below to update the WebGate profile created by the OAM CFG Tool.

1. Navigate to the Access System Console by specifying the following URL in your web browser:

```
http://hostname:port/access/oblix
```

where *hostname* refers to the host where WebPass Oracle HTTP Server instance is running and *port* refers to the HTTP port of the Oracle HTTP Server instance.

2. On the Access System main page, click the **Access System Console** link, then log in as an administrator.
3. On the Access System Console main page, click the **Access System Configuration** link to display the AccessGates Search page.
4. Enter the proper search criteria and click **Go** to display a list of AccessGates.
5. Select the AccessGate created by the OAM Configuration Tool. For example: WebCenter_EDG_AG).
6. On the AccessGate Details page, select **Modify** to display the Modify AccessGate page.
7. On the Modify AccessGate page, update:
 - **Hostname:** Update the hostname with the name of the computer where WebGate is running, for example: webhost1.mycompany.com.
 - **Preferred HTTP Host:** Update the Preferred_HTTP_Host with one of the hostname variations specified in the previous section, for example: admin.mycompany.com:7777.
 - **Primary HTTP Cookie Domain:** Update the Primary HTTP Cookie Domain with the Domain suffix of the host identifier, for example: mycompany.com
8. Click **Save**. A message box with the "Are you sure you want to commit these changes?" message is displayed.
9. Click **OK** to finish updating the configuration.
10. Verify the values displayed on the Details for AccessGate page to confirm that the updates were successful.

10.2.3.6 Adding Additional Access Servers

To assign an Access Server to the WebGate:

1. Log in as the Administrator on the Access System Console.
2. Navigate to the **Details** for AccessGate page, if necessary. From the Access System Console, select **Access System Configuration**, then **AccessGate Configuration**, then the link for the WebGate (**WebCenter_EDG_AG**).
3. On the **Details** for AccessGate page, click **List Access Servers**.
4. A page appears showing the primary or secondary Access Servers currently configured for this WebGate.
Click **Add**.
5. On the Add a New Access Server page, select an Access Server from the **Select Server** list, specify **Primary Server**, and define two connections for the WebGate.
Click the **Add** button to complete the association.

6. A page appears, showing the association of the Access Server with the WebGate. Click the link to display a summary and print this page for later use.
7. Repeat steps 3 through 6 to associate more Access Servers to the WebGate.

10.2.3.7 Configure Delegated Form Authentication

To configure the form authentication to redirect to the WebGate that was installed with the OAM installation, complete these steps:

1. Open the Access System Console.
2. In the Access System Configuration screen, select **Authentication Management** from the left-hand bar.
3. Select **OraDefaultFormAuthNScheme**.
4. Click **Modify**.
5. In the Challenge Redirect field, enter the host and port of the IDM installation; for example: `http://sso.mycompany.com`.

A WebGate should already be installed in the IDM installation. Refer to *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management* for details.

10.2.4 Installing and Configuring WebGate

WebGate needs to be installed on each of the WEBHOST n machines in order to secure the web tier:

1. Launch the WebGate installer (see [Section 1.5.5, "What to Install"](#) for information on where to obtain it) using the following command:

```
./Oracle_Access_Manager10_1_4_3_0_linux_OHS11g_WebGate -gui
```

2. The Welcome screen is displayed. Click **Next**.
3. In the Customer Information screen ([Figure 10–1](#)), enter the user name and user group that the web server is running as. Click **Next** to continue.

Figure 10–1 Customer Information Screen



Screenshot of the Customer Information screen.

4. In the installation target screen (Figure 10–2), specify the directory where WebGate should be installed. Click **Next** to continue.

Figure 10–2 Installation Target Screen



Screenshot of the installation target screen.

5. In the installation summary screen, click **Next**.

6. Download the required GCC runtime libraries for WebGate as instructed in the WebGate configuration screen (Figure 10-3), and use **Browse** to point to their location on the local computer. Click **Next** to continue.

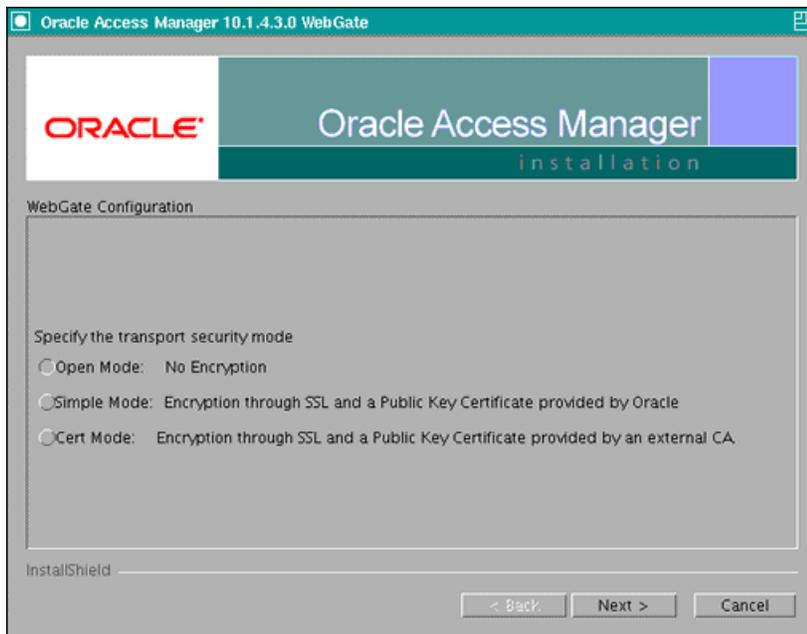
Figure 10-3 Runtime Libraries Screen



Screenshot of the WebGate configuration screen, which includes these instructions: "To proceed with installation of Oracle Access Manager 10.1.3.4.0 WebGate and for successfully running the product, you must install additional GCC runtime libraries, namely libgcc_s.so.1 and libstdc++.so.5. Note that these libraries should be compatible with GCC 3.3.2. The libraries are available for download from either of the following locations: <http://metalink.oracle.com> (requires login), or <http://www.oracle.com/technology/products/ias/index.html>. Once these libraries are locally available, please specify the directory containing the files and proceed with the installation."

7. The installer now creates the required artifacts. After that is completed, click **Next** to continue.
8. In the transport security mode screen (Figure 10-4), select "Open Mode: No Encryption" and click **Next** to continue.

Figure 10–4 Transport Security Mode Screen

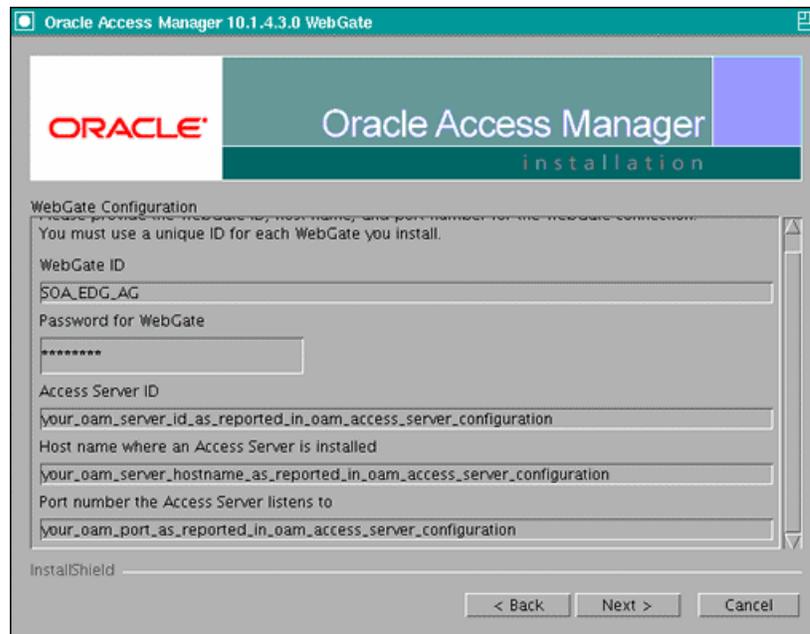


Screenshot of the transport security mode screen.

9. In the WebGate configuration screen, provide the details of the Access Server that will be used. You must provide the following information:
 - **WebGate ID**, as provided when the OAM configuration tool was executed
 - **Password for WebGate**
 - **Access Server ID**, as reported by the OAM Access Server configuration
 - **Access Server host name**, as reported by the OAM Access Server configuration
 - **Access Server port number**, as reported by the OAM Access Server configuration

Note: The Access Server ID, host name, and port are all required.

You can obtain these details from your Oracle Access Manager administrator. Click **Next** to continue.

Figure 10–5 Access Server Configuration Screen

Screenshot of the Access Server configuration screen.

10. In the Configure Web Server screen, click **Yes** to automatically update the web server. Click **Next** to continue.
11. In the next Configure Web Server screen, specify the full path of the directory containing the `httpd.conf` file. This file is located in the following directory:


```
ORACLE_BASE/admin/<OHS_Instance>/config/OHS/<OHS_ComponentName>
```

For example:

```
/u01/app/oracle/admin/ohs_instance2/config/OHS/ohs2/httpd.conf
```

Click **Next** to continue.
12. In the next Configure Web Server page, a message informs you that the Web server configuration has been modified for WebGate. Click **Yes** to confirm.
13. Stop and start your Web server for the configuration updates to take effect. Click **Next** to continue.
14. In the next Configure Web Server screen, the following message is displayed: "If the web server is set up in SSL mode, then the `httpd.conf` file needs to be configured with the SSL related parameters. To manually tune your SSL configuration, please follow the instructions that come up". Click **Next** to continue.
15. In the next Configure Web Server screen, a message with the location of the document that has information on the rest of the product setup and Web server configuration is displayed. Choose **No** and click **Next** to continue.
16. The final Configure Web Server screen appears with a message to manually launch a browser and open the HTML document for further information on configuring your Web server. Click **Next** to continue.

17. The Oracle COREid Readme screen appears. Review the information on the screen and click **Next** to continue.
18. A message appears (along with the details of the installation) informing you that the installation was successful.

10.2.5 Setting Up the WebLogic Authenticators

This section assumes that you have already set up the LDAP authenticator by following the steps in [Section 10.1.2.1, "Creating the LDAP Authenticator."](#) If you have not already created the LDAP authenticator, do it before continuing with this section.

This section includes the following topics:

- [Section 10.2.5.1, "Back Up Configuration Files"](#)
- [Section 10.2.5.2, "Setting Up the OAM ID Asserter"](#)
- [Section 10.2.5.3, "Setting the Order of Providers"](#)

10.2.5.1 Back Up Configuration Files

To be safe, first back up the relevant configuration files:

```
ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>/config/config.xml
ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>/config/fmwconfig/jps-con
fig.xml
ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>/config/fmwconfig/system-
jazn-data.xml
```

Also back up the `boot.properties` file for the Administration Server.

10.2.5.2 Setting Up the OAM ID Asserter

To set up the OAM ID Asserter, complete these steps:

1. Log into Weblogic Console, if not already logged in.
2. Navigate to `SecurityRealms\<Default Realm Name>\Providers`.
3. Click **New** and Select "OAM Identity Asserter" from the dropdown menu.
4. Name the asserter (for example, "OAM ID Asserter") and click **Save**.
5. Click the newly added asserter to see the configuration screen for OAM Identity Asserter.
6. Set the control flag to 'REQUIRED' and click **Save**.
7. Open the **Provider Specific** tab to configure the following required settings:
 - **Primary Access Server:** provide OAM server endpoint information in HOST:PORT format.
 - **AccessGate Name:** name of the AccessGate (for example, `WebCenter_EDG_AG`).
 - **AccessGate Password:** password for the AccessGate (optional).
8. Save the settings.

10.2.5.3 Setting the Order of Providers

Reorder the OAM Identity Asserter, OID Authenticator, and Default Authenticator by ensuring that the control flag for each authenticator is set as follows:

- OAM Identity Asserter: REQUIRED
- OID LDAP Authenticator (or OVD LDAP Authenticator): SUFFICIENT
- Default Authenticator: SUFFICIENT

10.2.6 Changing the Login Form for the Administration Console Application

Change the web.xml file for the console application to direct logins to the "/" URL. To accomplish this:

1. Make a backup of your WL_HOME/server/lib/consoleapp/webapp/WEB-INF/web.xml file:

```
SOAHOST1>cp WL_HOME/server/lib/consoleapp/webapp/WEB-INF/ web.xml WL_
HOME/server/lib/consoleapp/webapp/WEB-INF/ web.xml.backup
```

2. Edit the web.xml file and change the form-login-page URL to "/".

Specifically, change:

```
login-config>
  <auth-method>CLIENT-CERT,FORM</auth-method>
  <form-login-config>
    <form-login-page>/login/LoginForm.jsp</form-login-page>
    <form-error-page>/login/LoginError.jsp</form-error-page>
  </form-login-config>
</login-config>
```

to:

```
<login-config>
  <auth-method>CLIENT-CERT,FORM</auth-method>
  <form-login-config>
    <form-login-page>/</form-login-page>
    <form-error-page>/login/LoginError.jsp</form-error-page>
  </form-login-config>
</login-config>
```

3. To enable Administration Server failover with the same SSO behavior, repeat the above steps for the installation in SOAHOST2.
4. Restart the Administration Server.

10.3 Configuring WebCenter Applications

This section covers the following topics:

- [Section 10.3.1, "Configuring System Properties"](#)
- [Section 10.3.2, "Configuring the WebCenter Administrator Role"](#)
- [Section 10.3.3, "Configuring Discussion Server and Wiki for OAM"](#)

10.3.1 Configuring System Properties

There is a system property that tells WebCenter 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:

Table 10–1 System Property

Property	Value	Comment
oracle.webcenter.spaces.osso	true	This flag tells WebCenter that SSO is being used, so no login form should be displayed on the default landing page. Instead, it will render a login link that the user can click to invoke the SSO authentication.

To set this property, edit the `setDomainEnv.sh` script which is located in your `<adminserver_domain_home>/bin` directory. Add the property to the `EXTRA_JAVA_PROPERTIES` variable, as follows:

```
EXTRA_JAVA_PROPERTIES="-Dweblogic.security.SSL.ignoreHostnameVerification=true
-Doracle.mds.bypassCustRestrict=true -Djps.update.subject.dynamic=true
-Doracle.webcenter.spaces.osso=true -noverify ${EXTRA_JAVA_PROPERTIES}"
```

After making this change, restart the following servers:

- WebCenter's Administration Server
- All the domain's managed servers
- WebTier OHS

10.3.2 Configuring the WebCenter Administrator Role

After Oracle Internet Directory or Oracle Virtual Directory is configured as primary authenticator in WebCenter, the "weblogic" user should not be used as the WebCenter administrator. Create a user in Oracle Internet Directory and make that user the WebCenter administrator, either using WLST or Fusion Middleware Control:

- [Section 10.3.2.1, "Granting the WebCenter Spaces Administrator Role Using WLST"](#)
- [Section 10.3.2.2, "Granting the WebCenter Spaces Administrator Role Using Fusion Middleware Control"](#)

10.3.2.1 Granting the WebCenter Spaces Administrator Role Using WLST

To grant the WebCenter Administrator role using WLST:

1. Start WLST.
2. Connect to the WebCenter Spaces 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 Spaces administrator application role to the user in LDAP using the `grantAppRole` command as shown below:

```
grantAppRole(appStripe="webcenter", appRoleName="s8bba98ff_4cbb_40b8_bee_
```

```
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 in to WebCenter Spaces using the new account name. The Administration link should appear, and you should be able to perform all administrator operations.

10.3.2.2 Granting the WebCenter Spaces Administrator Role Using Fusion Middleware Control

This section describes how to grant the WebCenter Spaces administrator role to a user account other than the default "weblogic" account.

To grant the WebCenter Spaces Administrator role using Fusion Middleware Control:

1. Log into Fusion Middleware Control and select the WebLogic domain for WebCenter Spaces.
2. From the WebLogic Domain menu, select **Security**, and then **Application Roles**. The Application Roles page displays.
3. Search for the Administration application role by selecting the **Application** name for WebCenter Spaces (`WLS_Spaces/webcenter`), and providing the following internal identifier used by WebCenter Spaces as the **Role Name**:

```
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 name (`s8bba98ff_4cbb_40b8_beee_296c916a23ed#-#Administrator`) in the Role Name column.

The Edit Application Role page displays.

5. Click **Add User**.

The Add User pop-up displays.

6. Use the Search function to search for the user to assign the Administrator role to.
7. Use the arrow keys to move the user from the Available Users column to the Selected Users column, and click **OK**.
8. On the Edit Application Role page, click **OK**.
9. Restart the `WLS_Spaces` managed server.

When you log in to WebCenter Spaces, the Administration link should appear and you should be able to perform all administrator operations.

10.3.3 Configuring Discussion Server and Wiki for OAM

This section covers the following topics:

- [Section 10.3.3.1, "Setting Up Discussions Server to Use OAM as SSO Provider"](#)
- [Section 10.3.3.2, "Adding a Wiki Page to WebCenter"](#)

10.3.3.1 Setting Up Discussions Server to Use OAM as SSO Provider

To configure Oracle WebCenter Discussions Server for OAM single sign-on:

1. Log in to the Oracle WebCenter Discussions Server Admin Console at:

```
http://host:port/owc_discussions/admin
```

Where *host* and *port* are the host ID and port number of the **WLS_Services** managed server.
2. Open the System Properties page and edit, (if it already exists), or add the `owc_discussions.sso.mode` property, setting its value to `true`.
3. Edit or add the `jiveURL` property to point to the base URL of the SSO server. For example:

```
jiveURL = example.com:8890/owc_discussions
```

10.3.3.2 Adding a Wiki Page to WebCenter

The wiki page functionality is supported as a portlet that you can embed in a Web page (which also enables the SSO functionality). Since it does not require or support an identity store, there is no need to set up the LDAP store.

To add a wiki page to a WebCenter Group Space:

1. Log in to WebCenter Spaces, and navigate to a group space.
2. Add a page, choose **Web Page** as the style.
3. Once the page is created, click the **Edit** icon (pencil) on the top-right corner. In the Component Properties dialog, enter the following URL to the **Source** field:

```
http://<host>:<OHS port>/owc_wiki/page/show.jz?inline=1&scope=#{communityContext.communityName}
```

Note that it is the OHS port that is used, so that this will go through the WebGate which facilitates SSO.
4. Upon completion of specifying the component properties, the wiki page contents appear.
5. Save the changes.

10.4 Configuring WebCenter and BPEL Authentication

This section covers the following topics:

- [Section 10.4.1, "Set Authenticator"](#)
- [Section 10.4.2, "Set Role Members for BPMWorkflowAdmin Application Role in soa-infra"](#)

10.4.1 Set Authenticator

Ensure that the SOA domain is using the same authenticators as the WebCenter domain and has been configured for OAM Authentication.

10.4.2 Set Role Members for BPMWorkflowAdmin Application Role in soa-infra

When associating the domain with a identity store that does not contain the user "weblogic", you must assign some other valid user into the application role BPMWorkflowAdmin. To assign the role to a valid user, the following may be done:

1. Create a user in LDAP Store, in this case named WCAdmin, who will be assigned the role.
2. Assign the role. This can be done using `wlst` from the SOA Oracle home:

For example:

```
cd $ORACLE_HOME/common/bin/
wlst.sh

connect('weblogic','weblogic','SOADMINHOST:7001')
revokeAppRole(appStripe="soa-infra", appRoleName="BPMWorkflowAdmin",
principalClass="oracle.security.jps.service.policystore.ApplicationRole",
principalName="SOAdmin")
grantAppRole(appStripe="soa-infra", appRoleName="BPMWorkflowAdmin",
principalClass="weblogic.security.principal.WLSUserImpl",
principalName="WCAdmin")
```

10.5 Backing Up the Installation

After you have verified that the extended domain is working, back up the installation. This is a quick backup for the express purpose of immediate restore in case of problems in the further steps. The backup destination is the local disk. This backup can be discarded once the enterprise deployment setup is complete. At this point, the regular deployment-specific backup and recovery process can be initiated. The *Oracle Fusion Middleware Administrator's Guide* provides further details. For information on describing the Oracle HTTP Server data that must be backed up and restored, refer to the "Backup and Recovery Recommendations for Oracle HTTP Server" section in this guide. For information on how to recover components, see "Recovery of Components" and "Recovery After Loss of Component" sections in the guide. For recommendations specific to recovering from the loss of a host, see the "Recovering Oracle HTTP Server to a Different Host" in the guide. Also refer to the *Oracle Database Backup and Recovery User's Guide* for information on database backup.

To back up the installation at this point, complete these steps:

1. Back up the web tier:
 - a. Shut down the instance using `opmnctl`.

```
ORACLE_BASE/admin/<instance_name>/bin/opmnctl stopall
```
 - b. Back up the Middleware Home on the web tier using the following command (as root):

```
tar -cvpf BACKUP_LOCATION/web.tar $MW_HOME
```
 - c. Back up the Instance Home on the web tier using the following command (as root):

```
tar -cvpf BACKUP_LOCATION/web_instance.tar $ORACLE_INSTANCE
```
 - d. Start the instance using `opmnctl`:

```
ORACLE_BASE/admin/<instance_name>/bin/opmnctl startall
```
2. Back up the AdminServer domain directory. Perform a backup to save your domain configuration. The configuration files all exist under the `ORACLE_BASE/admin/<domain_name>` directory.

```
SOAHOST1> tar -cvpf edgdomainback.tar ORACLE_BASE/admin/<domain_name>
```

Managing the Topology

This chapter describes some operations that you can perform after you have set up the topology. These operations include monitoring, scaling, and backing up your topology.

This chapter contains the following sections:

- [Section 11.1, "Monitoring the Topology"](#)
- [Section 11.2, "Configuring UMS Drivers"](#)
- [Section 11.3, "Scaling the Topology"](#)
- [Section 11.4, "Performing Backups and Recoveries"](#)
- [Section 11.5, "Troubleshooting"](#)
- [Section 11.6, "Best Practices"](#)

11.1 Monitoring the Topology

For information on monitoring the topology, see chapter 15 of the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter*.

11.2 Configuring UMS Drivers

UMS driver configuration is not automatically propagated in a SOA cluster. This implies that users need to:

1. Apply the configuration of UMS drivers in each and every one of the servers in the EDG topology that is using the driver.
2. When server migration is used, servers are moved to a different node's domain directory. It is necessary to pre-create the UMS driver configuration in the failover node. The UMS driver configuration file location is:

```
ORACLE_BASE/admin/<domain_name>/msserver/<domain_name>/servers/<server_name>/  
tmp/_WL_user/<ums_driver_name>/*/configuration/driverconfig.xml
```

(where **** represents a directory whose name is randomly generated by WLS during deployment, for example, "3682yq").

In order to create the file in preparation for possible failovers, users can force a server migration and copy the file from the source node.

It is required to restart the driver for these changes to take effect (that is, for the driver to consume the modified configuration). To restart the driver:

1. Log on to the Oracle WebLogic Administration console.

2. Expand the environment node on the navigation tree.
3. Click on **Deployments**.
4. Select the driver.
5. Click **Stop->When work completes** and confirm the operation.
6. Wait for the driver to transition to the "Prepared" state (refresh the administration console page, if required).
7. Select the driver again, and click **Start->Servicing all requests** and confirm the operation.

Make sure that you verify in Oracle Enterprise Manager Fusion Middleware Control that the properties for the driver have been preserved.

11.3 Scaling the Topology

You can scale out and or scale up the enterprise topology. When you scale up the topology, you add new managed servers to nodes that are already running on one or more managed servers. When you scale out the topology, you add new managed servers to new nodes.

This section covers includes the topics:

- [Section 11.3.1, "Scaling Up the Topology \(Adding Managed Servers to Existing Nodes\)"](#)
- [Section 11.3.2, "Scaling Out the Topology \(Adding Managed Servers to New Nodes\)"](#)
- [Section 11.3.3, "Scaling Out the Oracle Content Server \(OCS\) Topology"](#)

11.3.1 Scaling Up the Topology (Adding Managed Servers to Existing Nodes)

In this case, you already have a node that runs a managed server configured with WebCenter components or a managed server with WSM-PM. The node contains a WebLogic Server home and an Oracle Fusion Middleware WebCenter home in shared storage.

You can use the existing installations (WebLogic Server home, Oracle Fusion Middleware home, and domain directories) for creating new Oracle WebCenter and WLS_WSM servers. There is no need to install WLS or WebCenter binaries in a new location, or to run pack and unpack.

Follow these steps for scaling up the topology:

1. Using the Administration Console, clone WLS_Spaces1 or WLS_Portlet1 or WLS_Services1 or WLS_WSM1 into a new managed server. The source managed server to clone should be one that already exists on the node where you want to run the new managed server.

To clone a managed server, complete these steps:

- a. Select **Environment** and then **Servers** from the Administration Console.
- b. Select the managed server to clone (for example, WLS_Spaces1 or WLS_Portlet1).
- c. Select **Clone**.

Name the new managed server (SERVER_NAME)n, where n is a number to identify the new managed server.

2. For the listen address, assign the host name or IP to use for this new managed server.
3. If the Managed Server being scaled out is WLS_Services, then follow these extra steps:
 - a. Start the Managed Server at least once, to create the Managed Servers deployment directories for Oracle Wiki. The Managed Server can then be shut down.
 - b. Configure the Jive server to be a member of the Jive Cluster by following the steps in [Section 6.14, "Converting Discussions Forum from Multicast to Unicast."](#) Use the following startup parameters.


```
-Dtangosol.coherence.wka1=Host1 -Dtangosol.coherence.wka2=Host2
-Dtangosol.coherence.localhost=Host1 -Dtangosol.coherence.wka1.port=8089
-Dtangosol.coherence.wka2.port=8089
```

Where Host1 is the Host where the new Managed Server has been created, and Host2 is any other Host in the cluster.
4. Reconfigure the Oracle HTTP Server module with the new member in the cluster. See Oracle HTTP Server configuration.
5. Add the New Managed Server to the Java Object Cache Cluster (see [Section 6.13, "Setting Up the Java Object Cache"](#)).

11.3.2 Scaling Out the Topology (Adding Managed Servers to New Nodes)

In scaling out your topology, you add new managed servers configured with Oracle WebCenter applications and or WSM-PM to new nodes.

Before performing the steps in this section, check that you meet these requirements:

- In your topology, there are existing nodes running managed servers configured with Oracle WebCenter applications and or WSM-PM.
- The new node can access the existing home directories for WebLogic Server and Oracle WebCenter. You use the existing installations in shared storage for creating a new managed server. You do not need to install WebLogic Server or WebCenter binaries in a new location but you do need to run pack and unpack to bootstrap the domain configuration in the new node.

Follow these steps for scaling out the topology:

1. On the new node, mount the existing FMW Home, which should include the WebCenter installation and the domain directory, and ensure that the new node has access to this directory, just like the rest of the nodes in the domain.
2. To attach ORACLE_HOME in shared storage to the local Oracle Inventory, execute the following command:

```
SOAHOSTn>cd ORACLE_HOME/
SOAHOSTn>./attachHome.sh -jreLoc ORACLE_BASE/fmw/jrockit_160_14_R27.6.4-18
```

To update the Middleware home list, create (or edit, if another WebLogic installation exists in the node) the `$HOME/boa/beahomelist` file and add `MW_HOME` to it.

3. Log in to the Oracle WebLogic Administration Console.
4. Create a new machine for the new node that will be used, and add the machine to the domain.

5. Update the machine's Node Manager's address to map the IP of the node that is being used for scale out.
6. Using the Administration Console, clone WLS_Spaces1/WLS_Portlet1/WLS_Services1/WLS_WSM1 into a new managed server and name it WLS_(SERVER_TYPE)*n*, where *n* is a number.

The steps assume that you are adding a new server to node *n*, where no managed server was running.
7. For the listen address of the managed server, assign the host name or IP to use for the new managed server.
8. If the Managed Server being scaled out is WLS_Services then follow these extra steps:
 - a. Start the Managed Server at least once, to create the Managed Servers deployment directories for Oracle Wiki. The Managed Server can then be shut down.
 - b. Configure the Jive server to be a member of the Jive Cluster by following the steps in [Section 6.14, "Converting Discussions Forum from Multicast to Unicast."](#) Use the following startup parameters.

```
-Dtangosol.coherence.wka1=Host1 -Dtangosol.coherence.wka2=Host2
-Dtangosol.coherence.localhost=Host1 -Dtangosol.coherence.wka1.port=8089
-Dtangosol.coherence.wka2.port=8089
```

Where Host1 is the Host where the new Managed Server has been created, and Host2 is any other Host in the cluster.

9. Reconfigure the Oracle HTTP Server module with the new member in the cluster (see Oracle HTTP Server configuration).
10. Start Node Manager on the new node: using the installation in shared storage from the already existing nodes, start Node Manager passing as parameter the host name of the new node:


```
NEW_NODE> WL_HOME/server/bin/startNodeManager <new_node_ip>
```

If you used the paths shown in [Section 4.1.1, "Installing Oracle WebLogic Server,"](#) WL_HOME would be ORACLE_BASE/wls.
11. Start and test the just added managed server from the Administration Console.

Access the application on the newly created managed server (<http://HOST:port/webcenter> or <http://host:port/wsm-pm>). The application should be functional.
12. Add the new managed server to the Java Object Cache Cluster (see [Section 6.13, "Setting Up the Java Object Cache"](#)).

11.3.3 Scaling Out the Oracle Content Server (OCS) Topology

Each Content Server node is installed using the cluster installer which can be found in `<content_server_dir>/bin` on the shared disk.

Installing Another Content Server Node

From the new node (SOAHOST*n*), run the command (all on one line):

```
Installer
-set-ClusterNodeIntradocDir=<ocs-stub>
-set-ClusterNodeName=<nodename>
```

```
-set-ClusterNodeAddress=<ip_of_node>
-set-ClusterBinDirRule=<local|shared>
ConfigureClusterNode
ConfigureAdminClusterNode
```

For <nodename>, specify **SOAHOSTn**.

For <stellent-stub>, use a local directory, for example, **/u01/app/oracle/product/ucm**.

For Set-ClusterBinDirRule, specify **local**.

Post-Installation Steps

After setting up the nodes, add the following lines to the `intradoc.cfg` file in the <stellent-stub>/bin and <stellent-stub>/admin/etc directories:

```
DisableSharedCacheChecking=true
ClusterNodeName=<nodename>
ClusterGroup=<cluster_name>
SocketServerAddress=<server_IP_address>
```

<nodename> is the identifier for the node.

<cluster_name> is the identifier for the cluster.

All nodes should have a different node name, but should have the same cluster group name; for example, the lines on SOAHOSTn might look like the following:

```
DisableSharedCacheChecking=true
ClusterNodeName=SOAHOSTn
ClusterGroup=ucmcluster
SocketServerAddress=<IP_of_SOAHOSTn>
```

Starting the Servers

Start Oracle Content Server and Administration Server on each node using the binaries in <ocs-stub>/bin.

Verify that the log and PID files are created in the <ocs-stub>/admin/etc and <ocs-stub>/etc directories.

Configuring the Load Balancer

You must reconfigure the load balancer to act as a socket load balancer to this new host as well. This will be used to make socket connections from the Oracle HTTP Server as well as WebCenter. Example:

Virtual host on load balancer: `wcinternal.mycompany.com:9054`

Maps to: `WCHOST1:9054, WCHOST2:9054, SOAHOSTn:9054`

Load-balancing method: `round-robin`

11.4 Performing Backups and Recoveries

[Table 11-1](#) lists the static artifacts to back up in the 11g Oracle WebCenter enterprise deployment.

Table 11–1 Static Artifacts to Back Up in the 11g Oracle WebCenter Enterprise Deployment

Type	Host	Location	Tier
ORACLE HOME (DB)	RAC Database hosts - CUSTDBHOST1 and CUSTDBHOST2	The location is user-defined	Directory Tier
MW HOME (SOA + WC)	SOAHOST1 and SOAHOST2 - SOA WCHOST1 and WCHOST2 - WC	MW_HOME on all hosts	Application Tier
ORACLE HOME (OHS)	WEBHOST1 and WEBHOST2	ORACLE_BASE/admin/<instance_name>	Web Tier
ORACLE HOME (OCS)	WCHOST1 and WCHOST2	On shared disk: /share/oracle/ucm On each host, local files at ORACLE_HOME/ucm	Application Tier
Installation-related files		OraInventory, <user_home>/bea/beahomelist, oraInst.loc, oratab	

Table 11–2 lists the runtime artifacts for back up in the 11g Oracle WebCenter enterprise deployment.

Table 11–2 Run-Time Artifacts to Back Up in the 11g Oracle WebCenter Enterprise Deployment

Type	Host	Location	Tier
DOMAIN HOME	SOAHOST1 SOAHOST2 WCHOST1 WCHOST2	ORACLE_BASE/admin/<domain_name>/mserver/<domain_name>	Application Tier
Application artifacts (ear and war files)	SOAHOST1 SOAHOST2 WCHOST1 WCHOST2	Look at all the deployments through admin console and get all the application artifacts	Application Tier
OHS INSTANCE HOME	WEBHOST1 and WEBHOST2	On WEBHOST1, ORACLE_HOME/ohs_1/instances/instance1 On WEBHOST2, ORACLE_HOME/ohs_2/instances/instance2	Web Tier
OHS OCS configuration files	WEBHOST1 and WEBHOST2	On each host, at /share/oracle/ucm, which is a local file system.	Web Tier
RAC databases	CUSTDBHOST1 and CUSTDBHOST2	The location is user-defined	Directory Tier
OCS content repository		Database-based	Directory Tier

For more information on backup and recovery of Oracle Fusion Middleware components, see *Oracle Fusion Middleware Administrator's Guide*.

Note: ORACLE_HOME should be backed up if any changes are made to the XEngine configuration that are part of your B2B setup. These files are located under ORACLE_HOME/soa/thirdparty/edifecs/XEngine. To back up ORACLE_HOME, execute the following command:

```
SOAHOST1> tar -cvpf fmwhomeback.tar MW_HOME
```

11.5 Troubleshooting

This section covers the following topics:

- [Section 11.5.1, "Administration Server Fails to Start After a Manual Failover"](#)
- [Section 11.5.2, "Error While Activating Changes in Administration Console"](#)
- [Section 11.5.3, "OAM Configuration Tool Does Not Remove URLs"](#)
- [Section 11.5.4, "Portlet Unavailable After Database Failover"](#)
- [Section 11.5.5, "Redirecting of Users to Login Screen After Activating Changes in Administration Console"](#)
- [Section 11.5.6, "Redirecting of Users to Administration Console's Home Page After Activating Changes to OAM"](#)
- [Section 11.5.7, "Configured JOC Port Already in Use"](#)
- [Section 11.5.8, "Restoring a JMS Configuration"](#)

11.5.1 Administration Server Fails to Start After a Manual Failover

Problem: Administration Server fails to start after the Administration Server node failed and manual failover to another nodes is performed. The Administration Server output log reports the following:

```
<Feb 19, 2009 3:43:05 AM PST> <Warning> <EmbeddedLDAP> <BEA-171520> <Could not obtain an exclusive lock for directory: ORACLE_BASE/admin/soadomain/aserver/soadomain/servers/AdminServer/data/ldap/ldapfiles. Waiting for 10 seconds and then retrying in case existing WebLogic Server is still shutting down.>
```

Solution: When restoring a node after a node crash and using shared storage for the domain directory, you may see this error in the log for the Administration Server due to unsuccessful lock cleanup. To resolve this error, remove the file `ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>/servers/AdminServer/data/ldap/ldapfiles/EmbeddedLDAP.lock`.

11.5.2 Error While Activating Changes in Administration Console

Problem: Activation of changes in Administration Console fails after changes to a server's start configuration have been performed. The Administration Console reports the following when clicking "Activate Changes":

```
An error occurred during activation of changes, please see the log for details.
[Management:141190]The commit phase of the configuration update failed with an exception:
In production mode, it's not allowed to set a clear text value to the property:
```

PasswordEncrypted of ServerStartMBean

Solution: This may happen when start parameters are changed for a server in the Administration Console. In this case, either provide username/password information in the server start configuration in the Administration Console for the specific server whose configuration was being changed, or remove the `<password-encrypted></password-encrypted>` entry in the `config.xml` file (this requires a restart of the Administration Server).

11.5.3 OAM Configuration Tool Does Not Remove URLs

Problem: The OAM Configuration Tool has been used and a set of URLs was added to the policies in Oracle Access Manager. One of multiple URLs had a typo. Executing the OAM Configuration Tool again with the correct URLs completes successfully; however, when accessing Policy Manager, the incorrect URL is still there.

Solution: The OAM Configuration Tool only adds new URLs to existing policies when executed with the same `app_domain` name. To remove a URL, use the Policy Manager Console in OAM. Log on to the Access Administration site for OAM, click on My Policy Domains, click on the created policy domain (SOA_EDG), then on the Resources tab, and remove the incorrect URLs.

11.5.4 Portlet Unavailable After Database Failover

Problem: While creating a page inside WebCenter Spaces, if you add a portlet to the page and a database failover occurs, an error component may be added to the page with the following message showing on it:

```
"Error"  
"Portlet unavailable"
```

This message remains even if you refresh the page or log out and back in again.

Solution: To resolve this issue, delete the component and add it again.

11.5.5 Redirecting of Users to Login Screen After Activating Changes in Administration Console

Problem: After configuring OHS and LBR to access the Oracle WebLogic Administration Console, some activation changes cause the redirection to the login screen for the admin console.

Solution: This is the result of the console attempting to follow changes to port, channel, and security settings as a user makes these changes. For certain changes, the console may redirect to the Administration Server's listen address. Activation is completed regardless of the redirection. It is not required to log in again; users can simply update the URL to `soa.mycompany.com/console/console.portal` and directly access the home page for the Administration Console.

Note: This problem will not occur if you have disabled tracking of the changes described in this section.

11.5.6 Redirecting of Users to Administration Console's Home Page After Activating Changes to OAM

Problem: After configuring OAM, some activation changes cause the redirection to the Administration Console's home page (instead of the context menu where the activation was performed).

Solution: This is expected when OAM SSO is configured and is the result of the redirections performed by the Administration Server. Activation is completed regardless of the redirection. If required, users may "manually" navigate again to the desired context menu.

11.5.7 Configured JOC Port Already in Use

Problem: Attempts to start a Managed Server that uses the Java Object Cache, such as OWSM or WebCenter Spaces Managed Servers, fail. The following errors appear in the logs:

```
J2EE JOC-058 distributed cache initialization failure
J2EE JOC-043 base exception:
J2EE JOC-803 unexpected EOF during read.
```

Solution: Another process is using the same port that JOC is attempting to obtain. Either stop that process, or reconfigure JOC for this cluster to use another port in the recommended port range.

11.5.8 Restoring a JMS Configuration

Problem: A mistake in the parameters passed to the `soa-createUDD.py` script, or some other mistake causes the JMS configuration for SOA or BAM clusters to fail.

Solution: Use `soa-createUDD.py` to restore the configuration.

If a mistake is made while running the `soa-createUDD.py` script after the SOA cluster is created from the Oracle Fusion Middleware Configuration Wizard (an incorrect option is used, a target is modified, or a module is deleted accidentally). In these situations you can use the `soa-createUDD.py` script to restore the appropriate JMS configuration using the following steps:

1. Delete the existing SOA JMS resources (JMS Modules owned by the `soa-infrastructure` system).
2. Run the `soa-createUDD.py` again. The script assume the JMS Servers created for SOA are preserved and creates the destinations and subdeployment modules required to use Uniform Distributed Destinations for SOA. In this case, the script should be executed with the option `--soacluster`. After running the script again, verified from the WebLogic Server Administration Console that the following artifacts exist (**Domain Structure, Services, Messaging, JMS Modules**):

```
SOAJMSModuleUDDs          ---->SOAJMSSubDM targeted to SOAJMSServer_auto_1 and
SOAJMSServer_auto_2
UMSJMSSystemResource     ---->UMSJMSSubDMSOA targeted to UMSJMSServer_auto_1 and
UMSJMSServer_auto_2
```

11.6 Best Practices

This section covers the following topics:

- [Section 11.6.1, "Preventing Timeouts for SQLNet Connections"](#)

- [Section 11.6.2, "Auditing"](#)

11.6.1 Preventing Timeouts for SQLNet Connections

Much of the EDG production deployment involves firewalls. Because database connections are made across firewalls, Oracle recommends that the firewall be configured so that the database connection is not timed out. For Oracle Real Application Clusters (RAC), the database connections are made on Oracle RAC VIPs and the database listener port. You must configure the firewall to not time out such connections. If such a configuration is not possible, set the `*SQLNET.EXPIRE_TIME=n*` parameter in the `ORACLE_HOME/network/admin/sqlnet.ora` file on the database server, where `n` is the time in minutes. Set this value to less than the known value of the timeout for the network device (that is, a firewall). For RAC, set this parameter in all of the Oracle home directories.

11.6.2 Auditing

Oracle Fusion Middleware Audit Framework is a new service in Oracle Fusion Middleware 11g, designed to provide a centralized audit framework for the middleware family of products. The framework provides audit service for platform components such as Oracle Platform Security Services (OPSS) and Oracle Web Services. It also provides a framework for JavaEE applications, starting with Oracle's own JavaEE components. JavaEE applications will be able to create application-specific audit events. For non-JavaEE Oracle components in the middleware such as C or JavaSE components, the audit framework also provides an end-to-end structure similar to that for JavaEE applications.

[Figure 11-1](#) is a high-level architectural diagram of the Oracle Fusion Middleware Audit Framework.

Figure 11-1 Audit Event Flow

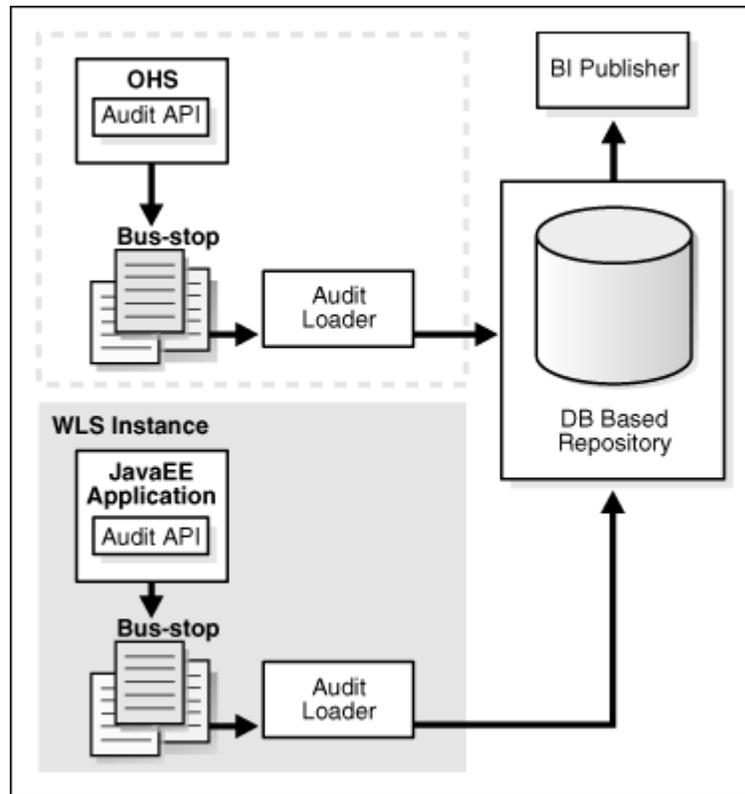


Diagram representation of the audit event flow, which is described in the text following this diagram.

The Oracle Fusion Middleware Audit Framework consists of the following key components:

- **Audit APIs**

These are APIs provided by the audit framework for any audit-aware components integrating with the Oracle Fusion Middleware Audit Framework. During runtime, applications may call these APIs where appropriate to audit the necessary information about a particular event happening in the application code. The interface allows applications to specify event details such as username and other attributes needed to provide the context of the event being audited.

- **Audit Events and Configuration**

The Oracle Fusion Middleware Audit Framework provides a set of generic events for convenient mapping to application audit events. Some of these include common events such as authentication. The framework also allows applications to define application-specific events.

These event definitions and configurations are implemented as part of the audit service in Oracle Platform Security Services. Configurations can be updated through Enterprise Manager (UI) and WLST (command-line tool).

- **Audit Bus-stop**

Bus-stops are local files containing audit data before they are pushed to the audit repository. In the event where no database repository is configured, these bus-stop

files can be used as a file-based audit repository. The bus-stop files are simple text files that can be queried easily to look up specific audit events. When a DB-based repository is in place, the bus-stop acts as an intermediary between the component and the audit repository. The local files are periodically uploaded to the audit repository based on a configurable time interval.

- **Audit Loader**

As the name implies, audit loader loads the files from the audit bus-stop into the audit repository. In the case of platform and JavaEE application audit, the audit loader is started as part of the JavaEE container start-up. In the case of system components, the audit loader is a periodically spawned process.

- **Audit Repository**

Audit Repository contains a pre-defined Oracle Fusion Middleware Audit Framework schema, created by Repository Creation Utility (RCU). Once configured, all the audit loaders are aware of the repository and upload data to it periodically. The audit data in the audit repository is expected to be cumulative and will grow overtime. Ideally, this should not be an operational database used by any other applications - rather, it should be a standalone RDBMS used for audit purposes only. In a highly available configuration, Oracle recommends that you use an Oracle Real Application Clusters (RAC) database as the audit data store.

- **Oracle Business Intelligence Publisher**

The data in the audit repository is exposed through pre-defined reports in Oracle Business Intelligence Publisher. The reports allow users to drill down the audit data based on various criteria. For example:

- Username
- Time Range
- Application Type
- Execution Context Identifier (ECID)

For more introductory information for the Oracle Fusion Middleware Audit Framework, see the "Introduction to Oracle Fusion Middleware Audit Framework" chapter in the *Oracle Fusion Middleware Security Guide*.

For information on how to configure the repository for Oracle Fusion Middleware Audit Framework, see the "Configuring and Managing Auditing" chapter in the *Oracle Fusion Middleware Security Guide*.

The EDG topology does not include Oracle Fusion Middleware Audit Framework configuration. The ability to generate audit data to the bus-stop files and the configuration of the audit loader will be available once the products are installed. The main consideration is the audit database repository where the audit data is stored. Because of the volume and the historical nature of the audit data, it is strongly recommended that customers use a separate database from the operational store or stores being used for other middleware components.

A

access gate, 10-12
adding clusters, 5-6, 6-4
adding managed servers, 5-6, 6-4
adding managed servers to existing nodes, 11-2
adding managed servers to new nodes, 11-3
Administration Console
 frontend URL, 4-17
 login form, 10-21
 redirecting to home page, 11-9
 redirecting to login screen, 11-8
administration server, 4-9, 4-10, 4-16
 failover, 4-19, 4-20
 host name verification, 4-11, 4-13
 restarting, 5-8, 6-6
 SSL communication, 7-1, 7-5
 starting, 4-10
 validating, 4-10
administrator role for WebCenter, 10-22
admin.mycompany.com, 2-6
application tier, 1-9
arping, 4-4
ASM, see 'Automatic Storage Management (ASM)'
assigning servers to clusters, 5-6, 6-4
assigning servers to machines, 5-7, 6-5
audit APIs, 11-11
audit bus-stops, 11-11
audit events, 11-11
Audit Framework, 11-10
audit loader, 11-12
audit repository, 11-12
auditing, 11-10
authenticators, 10-20, 10-24
Automatic Storage Management (ASM), 2-2

B

backups
 configuration files, 10-2, 10-20
 database, 2-5
 domain, 5-3, 6-2
 enterprise deployments, 11-5
 installation, 4-3, 4-21, 5-21, 6-17, 10-25
 Oracle Content Server, 9-8, 9-17
 Oracle HTTP Server, 3-4

best practices
 auditing, 11-10
 timeouts for SQLNet connections, 11-10
boot.properties, 4-10
BPEL authentication, 10-24
BPMWorkflowAdmin application role, 10-24
built-in security, 1-5
bus-stops, 11-11

C

cache for Java objects, 6-9
callback URL, 5-16
cluster agent, 1-3
clusters, 1-2, 4-8
 adding, 5-6, 6-4
 assigning servers, 5-6, 6-4
 Oracle Content Server, 9-14
clusterware, 1-3
Coherence, see 'Oracle Coherence'
configuration
 cluster for Oracle Content Server, 9-14
 database, 2-1
 delegated form authentication, 10-15
 directory structure, 2-11
 discussion forum, 8-1
 document library, 8-4
 domain on SOAHOST1, 4-4
 frontend HTTP host and port, 5-16
 high availability for Oracle File and FTP
 Adapters, 5-18
 Instant Messaging and Presence (IMP), 8-2
 load balancer, 2-6
 load balancer for Oracle Content Server, 9-14,
 9-16
 network, 2-5
 Oracle Coherence, 5-8
 Oracle Content Server for Remote Oracle HTTP
 Server, 9-6, 9-16
 Oracle Content Server for WebCenter, 9-8
 Oracle HTTP Server, 4-16
 Oracle HTTP Server for WLS managed
 servers, 5-14, 6-15
 persistence store for transaction recovery, 5-17
 portlet producers, 8-5
 RAC for Oracle Content Server, 9-17

- scaling Oracle Database Adapter, 5-21
- shared JMS persistence store, 5-17
- shared storage, 2-11, 2-17
- UMS drivers, 11-1
- use of custom keystores, 7-4, 7-7
- WebCenter and Oracle Content Server, 9-5
- WebCenter applications for OAM, 10-21
- WebGate, 10-15
- wiki, 10-24
- wiki server, 6-14
- Workflow, 8-3
- Worklist, 8-3
- Configuration Wizard, 4-4, 5-3, 6-2
- Configure JDBC Component Schema screen, 5-4
- Configure RAC Multi Data Source Component Schema screen, 5-4
- configure-joc.py script, 4-14, 6-9
- connection factory parameters, 5-19
- connections
 - discussion forum, 8-1
 - document library, 8-4
 - Instant Messaging and Presence (IMP), 8-2
 - Workflow, 8-3
 - Worklist, 8-3
- content server cluster, 9-14
- Content Server, see 'Oracle Content Server'
- CREATE_SERVICE, 2-3
- creating identity keystore, 7-2, 7-6
- creating trust keystore, 7-3, 7-6
- CUSTDBHOST nodes, 1-10, 2-2
- custom keystores, 7-4, 7-7

D

- data sources, 4-6
- data tier, 1-10
- database
 - backing up, 2-5
 - CREATE_SERVICE, 2-3
 - host requirements, 2-2
 - initialization parameters, 2-2
 - loading repository, 2-4
 - mutex locking, 5-18
 - services, 2-3
 - setting up, 2-1
 - supported versions, 2-2
- database listener port, 1-10
- database preconfiguration, 2-1
- debug level for Oracle Content Server, 9-4
- default persistence store for transaction recovery, 5-17
- delegated form authentication, 10-15
- directory structure, 2-11, 2-12, 2-16
- disabling host name verification, 4-11, 4-13, 5-11, 6-6
- Discussion Forum, 1-9
- discussion forum, 8-1
- DMZ, 1-5, 1-8, 1-9
- document library, 8-4
- domain
 - backing up, 5-3, 6-2

- creating on SOAHOST1, 4-4
- extending for SOA components, 5-1, 5-3
- extending for WebCenter components, 6-1, 6-2
- domain configuration
 - propagating, 4-13, 5-12, 5-13, 6-6, 6-7
- DOMAIN directory, 2-11
- domain directory, 4-9

E

- enabling SOAHOST1VHN1 on SOAHOST1, 4-4
- enterprise deployment, 1-1
 - backups and recoveries, 11-5
 - topology, 1-6
- extending domain
 - for SOA components, 5-1, 5-3
 - for WebCenter components, 6-1, 6-2
- external communication, 1-5
- external services
 - discussion forum, 8-1
 - document library, 8-4
 - Instant Messaging and Presence (IMP), 8-2
 - portlet producers, 8-5
 - Worklist, 8-3
- EXTRA_JAVA_PROPERTIES, 10-22

F

- failback, 1-2
- failover, 1-2, 11-7
- failover of administration server, 4-19, 4-20
- firewalls, 2-9
- form-login-page URL, 10-21
- frontend HTTP host and port, 5-16
- frontend URL for Administration Console, 4-17
- Fusion Middleware Audit Framework, 11-10
- Fusion Middleware, see 'Oracle Fusion Middleware'

G

- generating self-signed certificates, 7-2, 7-5
- granting the WebCenter administrator role, 10-22
- grid servers, 1-1

H

- hardware cluster, 1-2
- hardware requirements, 1-6
- high availability, 1-1, 1-6, 5-8
 - Oracle File and FTP Adapters, 5-18
- home page, redirecting to, 11-9
- host identifier, 10-13
- host name, 5-9
 - network, 1-3
 - physical, 1-3
 - virtual, 1-4
- host name verification, 4-11, 4-13, 5-11, 6-6
- HTTP port, 1-8
- httpd.conf, 3-3, 4-16, 6-16
- HTTPS port, 1-8

I

ID Asserter, 10-20
IdcServer, 9-4
identity keystore, 7-2, 7-6
ifconfig, 4-4
IMP, see 'Instant Messaging and Presence (IMP)'
incorrect URLs, 11-8
initialization parameters for database, 2-2
installation
 Oracle Content Server, 9-1
 Oracle Fusion Middleware, 4-3, 5-2, 6-2
 Oracle Fusion Middleware Home, 4-2
 Oracle HTTP Server, 3-1
 Oracle WebLogic Server, 4-2
 procedure, 1-11
 strategies, 1-12
 WebGate, 10-15
 what to install, 1-10
Instant Messaging and Presence (IMP), 8-2
IPs, 2-8

J

Java object cache, 6-9
JDBC component schema, 5-4
JDK, 4-2
JMS persistence store, 5-17
JRockit, 4-2

K

keystores
 custom, 7-4, 7-7
 identity, 7-2, 7-6
 trust, 7-3, 7-6
keytool utility, 7-3, 7-6

L

LDAP
 moving WebLogic administrator to --, 10-4
load balancer, 1-8, 3-3
 configuration, 2-6
 Oracle Content Server, 9-14, 9-16
 requirements, 1-8
 validating access, 6-17
locations of directories, 2-12, 2-16
log files for Oracle Content Server, 9-4
login form for Administration Console, 10-21
login screen, redirecting to, 11-8

M

managed servers, 4-8, 4-9
 adding, 5-6, 6-4
 adding to existing nodes, 11-2
 adding to new nodes, 11-3
 propagating domain changes, 5-12, 6-6
 validation, 5-12, 5-13
 WebCenter, 6-6

 WLS_Portlet, 6-8, 6-9, 6-15
 WLS_Services, 6-8, 6-9, 6-15
 WLS_SOA, 5-11, 5-12, 5-13
 WLS_Spaces, 6-8, 6-9, 6-15
 WLS_WSM, 4-10, 4-11, 4-12, 4-13, 4-14, 4-16
managing the topology, 11-1
manual failover, 11-7
manual failover of administration server, 4-19
mapping of IPs and VIPs, 2-8
Middleware home, 1-2
mod_wl_ohs.conf file, 4-16, 6-15
monitoring the topology, 11-1
mutex locking, 5-18
MW_HOME, 2-11

N

names of virtual servers, 2-5
network
 firewalls, 2-9
 IPs, 2-8
 load balancers, 2-6
 ports, 2-9
 shared storage, 2-15
 virtual IPs (VIPs), 2-8
 virtual servers, 2-5
network host name, 1-3
network preconfiguration, 2-5
Node Manager
 restarting, 5-11, 5-13
 setup, 7-1
 SSL communication, 7-1, 7-5
 starting, 4-12, 4-14, 6-6, 6-7, 7-4, 7-8
 use of custom keystores, 7-4, 7-7
nodes
 adding servers to existing --, 11-2
 adding servers to news --, 11-3
 application tier, 1-9
 CUSTDBHOST, 1-10, 2-2
 data tier, 1-10
 primary, 1-3
 secondary, 1-3
 SOAHOST, 4-4, 4-10, 4-12, 4-14, 5-2, 5-11, 5-13,
 6-2, 6-6
 WCHOST, 1-9, 6-6, 6-7
 web tier, 1-8
 WEBHOST, 1-8, 3-1
nodes of content servers, 9-14

O

OAM, see 'Oracle Access Manager (OAM)'
OAMCFG tool, 11-8
 overview, 10-10, 10-11
 running, 10-11
OAP port, 1-10
OCS schema, 9-1
OCS, see 'Oracle Content Server (OCS)'
OID authenticator, 10-2
OID ports, 1-10

- OmniPortlet URL, 8-5
- Oracle Access Manager, 1-8
- Oracle Access Manager (OAM)
 - BPEL authentication, 10-24
 - configuring WebCenter applications, 10-21
 - delegated form authentication, 10-15
 - ID Asserter, 10-20
 - OAMCFG tool, 10-10, 10-11
 - order of providers, 10-20
 - overview, 10-10
 - prerequisites, 10-10
 - updating host identifier, 10-13
 - updating WebGate profile, 10-13
 - verifying access gate, 10-12
 - verifying policy domain, 10-12
 - WebGate, 10-15
 - WebLogic authenticators, 10-20
- Oracle Access Protocol (OAP), 1-8
- Oracle Business Intelligence Publisher, 11-12
- Oracle Coherence, 5-8
 - enabling unicast communication, 5-8
 - specifying host name, 5-9
- Oracle Content Server
 - backup, 9-8, 9-17
 - cluster, 9-14
 - configuration, 9-8
 - configuration for remote Oracle HTTP Server, 9-6, 9-16
 - configuration for WebCenter, 9-8
 - configuring identity store, 9-8
 - configuring SSL, 9-11
 - debug level, 9-4
 - enabling search and indexing, 9-10
 - HTTP address, 9-16
 - installation, 9-1
 - load balancer, 9-14, 9-16
 - nodes, 9-14
 - prerequisites
 - RAC, 9-17
 - starting, 9-4
 - updating, 9-5
 - validation, 9-8
 - verifying signatures, 9-12
 - WcConfigure component, 9-5
- Oracle Content Server (OCS)
 - scaling out the topology, 11-4
- Oracle Content Servers
 - log files, 9-4
- Oracle Database Adapter, scaling, 5-21
- Oracle File and FTP Adapters, 5-18
- Oracle Fusion Middleware
 - installation, 4-3, 5-2, 6-2
 - installing Home, 4-2
 - installing Oracle WebLogic Server, 4-2
- Oracle Fusion Middleware Audit Framework, 11-10
- Oracle Fusion Middleware Configuration Wizard, 4-4
- Oracle home, 1-2
- Oracle HTTP Server
 - backup, 3-4

- configuration, 4-16
- configuration for WLS managed servers, 6-15
- and Content Server, 9-6, 9-16
- installation, 3-1
- registering, 4-17
- validating access, 4-18, 4-20, 5-15, 6-17
- validation, 3-3

- Oracle instance, 1-2
- Oracle WebLogic Server
 - installation, 4-2
 - registering Oracle HTTP Server, 4-17
- ORACLE_BASE, 2-11
- ORACLE_HOME, 2-11
- ORACLE_INSTANCE, 2-12

P

- parameters for connection factory, 5-19
- performance, enterprise deployment and, 1-1
- persistence store
 - shared JMS, 5-17
 - transaction recovery, 5-17
- physical host name, 1-3
- physical IP, 1-4
- policy domain, 10-12
- portlet producers, 8-5
- portlets, 1-9, 11-8
- ports
 - database listener, 1-10
 - frontend HTTP, 5-16
 - HTTP, 1-8
 - HTTPS, 1-8
 - Oracle HTTP Server, 3-1
 - Oracle Internet Directory (OID), 1-10
 - used in topology, 2-9
- preconfiguration
 - database, 2-1
 - directory structure, 2-11
 - network, 2-5
 - shared storage, 2-11
- primary node, 1-3
- PROCESSES parameter for database, 2-2
- propagating domain changes, 5-12, 6-6
- propagating domain configuration, 4-13, 5-13, 6-7
- provider order for OAM, 10-20

Q

- Quartz, 5-3

R

- RAC and Oracle Content Server, 9-17
- RAC database, 1-10, 4-6
- RAC multi-data source component schema, 5-4
- recovery of enterprise deployments, 11-5
- redirecting to home page, 11-9
- redirecting to login screen, 11-8
- reference topology, 1-6
- registering Oracle HTTP Server, 4-17
- registering portlet producers, 8-5

- remote Oracle HTTP Server, 9-6, 9-16
- Repository Creation Utility (RCU), 2-1, 2-4
- requirements
 - database host, 2-2
 - load balancer, 1-8
- requirements, hardware, 1-6
- restarting administration server, 5-8, 6-6
- restarting Node Manager, 5-11, 5-13
- running the soa-createUDD.py script, 5-10

S

- scaling Oracle Database Adapter, 5-21
- scaling out the OCS topology, 11-4
- scaling out the topology, 11-3
- scaling the topology
 - topology
 - scaling, 11-2
- scaling up the topology, 11-2
- screens
 - Configure JDBC Component Schema, 5-4
 - Configure RAC Multi Data Source Component Schema, 5-4
- scripts
 - configure-joc.py, 4-14, 6-9
 - setDomainEnv.sh, 5-10, 10-22
 - setNMProps.sh, 4-12, 4-14
 - soa-createUDD.py, 5-10
 - wc_contentserverconfig.sh, 9-5
- secondary node, 1-3
- security, 1-5
- self-signed certificates, 7-2, 7-5
- servers, 4-8
 - assigning to clusters, 5-6, 6-4
 - assigning to machines, 5-7, 6-5
- service level agreements, 1-1
- setDomainEnv.sh script, 5-10, 10-22
- setNMProps.sh script, 4-12, 4-14
- setting up Java object cache, 6-9
- setting up Node Manager, 7-1
- setting up WebLogic authenticators, 10-20
- shared JMS persistence store, 5-17
- shared storage, 1-3, 2-11, 2-15
 - configuration, 2-17
- SOA Home, 5-2, 6-2
- soa-createUDD.py script, 5-10
- SOAHOST nodes, 4-4, 4-10, 4-12, 4-14, 5-2, 5-11, 5-13, 6-2, 6-6
- SOAHOST1VHn virtual hosts, 5-9
- soainternal.mycompany.com, 2-6
- SQLNet connections, timeouts, 11-10
- SSL
 - Oracle Content Server, 9-11
- SSL acceleration, 1-9
- SSL communication, 7-1, 7-5
- SSO mode, 10-21
- starting administration server, 4-10
- starting Node Manager, 4-12, 4-14, 6-6, 6-7, 7-4, 7-8
- starting Oracle Content Server, 9-4
- starting WLC_Services managed server, 6-8, 6-9

- starting WLS_Portlet managed server, 6-8, 6-9
- starting WLS_SOA managed server, 5-12, 5-13
- starting WLS_Spaces managed server, 6-8, 6-9
- starting WLS_WSM managed server, 4-12, 4-14
- staticports.ini, 3-2
- storage, 2-11, 2-15
- strategies for installation, 1-12
- supported database versions, 2-2
- switchback, 1-4
- switchover, 1-4

T

- targeted applications, 5-7
- targeting deployments, 4-9
- timeouts for SQLNet connections, 11-10
- topology, 1-6
 - application tier, 1-9
 - data tier, 1-10
 - database, 2-1
 - directory structure, 2-11
 - managing, 11-1
 - monitoring, 11-1
 - network, 2-5
 - scaling out, 11-3
 - scaling out (OCS), 11-4
 - scaling up, 11-2
 - shared storage, 2-11
 - web tier, 1-8
- transaction recovery, 5-17
- troubleshooting
 - activating changes in Admin Server, 11-7
 - incorrect URLs, 11-8
 - manual failover, 11-7
 - portlet unavailable, 11-8
 - redirecting to home page, 11-9
 - redirecting to login screen, 11-8
- trust keystore, 7-3, 7-6

U

- UCM, see 'Universal Content Management'
- UMS drivers, 11-1
- unicast communication, 1-10, 5-8
- Universal Content Management (UCM), 1-9
 - scaling out OCS topology, 11-4
- unpack utility, 4-13, 5-13, 6-7
- updating the host identifier, 10-13
- updating WebGate profile, 10-13
- URL, callback, 5-16
- utils.CertGen utility, 7-2, 7-5
- utils.ImportPrivateKey utility, 7-2, 7-6

V

- validating an Oracle Content Server installation, 9-8
- validation
 - access through load balancer, 6-17
 - access through Oracle HTTP Server, 4-18, 4-20, 5-15, 6-17
 - administration server, 4-10

- Oracle HTTP Server, 3-3
- WLC_Services managed server, 6-8, 6-9
- WLS_Portlet managed server, 6-8, 6-9
- WLS_SOA managed server, 5-12, 5-13
- WLS_Spaces managed server, 6-8, 6-9
- WLS_WSM managed server, 4-12, 4-14, 5-12
- verification of host names, 4-11, 4-13, 5-11, 6-6
- VIPs, 2-8
 - enabling SOAHOST1VHN1 on SOAHOST1, 4-4
- virtual host name, 1-4
- virtual IP, 1-4
- virtual IPs (VIPs), 2-8
- virtual server names, 2-5
- virtual servers, 1-9
 - admin.mycompany.com, 2-6
 - soainternal.mycompany.com, 2-6
 - wcinternal.mycompany.com, 2-6
 - wc.mycompany.com, 2-5
- <VirtualHost> entries in httpd.conf, 3-3, 4-16, 6-16

W

- wc_contentserverconfig.sh, 9-5
- WcConfigure component, 9-5
- WCHOST nodes, 1-9, 6-6, 6-7
- wcinternal.mycompany.com, 2-6
- wc.mycompany.com, 2-5
- web tier, 1-8
- WebCenter
 - administrator role, 10-22
 - authentication, 10-24
 - configuration of Oracle Content Server, 9-8
 - configuration script, 9-5
 - configuring applications for OAM, 10-21
 - disabling host name verification, 6-6
 - extending domain for --, 6-1, 6-2
 - installing Oracle Fusion Middleware, 4-3
 - SSO mode, 10-21
 - wiki server, 6-14
- WebCenter components, 1-9
- WebCenter Spaces, 1-9, 6-8, 6-9
 - discussion forum, 8-1
 - document library, 8-4
 - Instant Messaging and Presence (IMP), 8-3
 - Java object cache, 6-9
 - portlet producers, 8-5
 - Workflow, 8-3
 - Worklist, 8-3
- WebClipping URL, 8-5
- WebGate, 1-8, 10-15
- WebGate profile, 10-13
- WEBHOST nodes, 1-8, 3-1
- WebLogic administrator, moving to LDAP, 10-4
- WebLogic authenticators, 10-20
- WebLogic Configuration Wizard, 4-4
- WebLogic Scripting Tool (WLST)
 - discussion forum, 8-1
 - document library, 8-5
 - Instant Messaging and Presence (IMP), 8-3
 - portlet producers, 8-5

- Workflow, 8-4
- Worklist, 8-4
- WebLogic Server home, 1-2
- WebLogic Server, see 'Oracle WebLogic Server' Wiki, 1-9
- wiki server, 6-14, 10-24
- WL_HOME, 2-11
- WLS_Portlet managed server, 6-8, 6-9, 6-15
- WLS_Services managed server, 6-8, 6-9, 6-15
- WLS_SOA
 - disabling host name verification, 5-11
- WLS_Spaces managed server, 6-8, 6-9, 6-15
- WLS_WSM, 4-10, 4-16
 - disabling host name verification, 4-11, 4-13
 - starting, 4-12, 4-14
 - validating, 4-12, 4-14
- WLST, see 'WebLogic Scripting Tool'
- Workflow, 8-3
- Worklist, 8-3

X

- XEngine, 5-13