Oracle® Fusion Middleware

Enterprise Deployment Guide for Oracle WebCenter Content

11*g* Release 1 (11.1.1)

E15483-09

March 2013

Documentation for installers that describes how to install and configure Oracle WebCenter Content components in an enterprise deployment. Includes best practices blueprint for an Oracle WebCenter Content enterprise deployment topology.

ORACLE

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

E15483-09

Primary Authors: Ron van de Crommert (Writer), Bonnie Vaughan (Writer), Janga Aliminati (Architect), Fermin Castro (Contributing Engineer), Lingaraj Nayak (Contributing Engineer)

Contributing Author: Joe Paul

Contributors: Pradeep Bhat, Richard Delval, Eileen He, Van Sioung Ng Yan Tun, Madhwa Chintarevula, Izeta Delic, Nouar Garcia-Mardambek, Pascal Prevot

Contents

3 Preparing the Network for an Enterprise Deployment

4 Preparing the File System for an Enterprise Deployment

5 Preparing the Database for an Enterprise Deployment

6 Installing the Software for an Enterprise Deployment

7 Configuring Oracle Web Tier

8 Creating a Domain for an Enterprise Deployment

9 Extending the Domain to Include Oracle SOA Suite Components

10 Extending the Domain to Include Oracle WebCenter Content

11 Extending the Domain to Include Imaging

12 Extending the Domain to Include Inbound Refinery

13 Setting Up Node Manager

16 Managing the Topology

Preface

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

Audience

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

Documentation Accessibility

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

Access to Oracle Support

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

Related Documents

This guide refers to the following Oracle Fusion Middleware documents for more information:

- *Oracle Fusion Middleware Administering Oracle WebCenter Content*
- *Oracle Fusion Middleware Administrator's Guide*
- *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager with Oracle Security Token Service*
- *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite*
- *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server*
- *Oracle Fusion Middleware Configuring Server Environments for Oracle WebLogic Server*
- *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*
- *Oracle Fusion Middleware High Availability Guide*

- *Oracle Fusion Middleware Installation Guide for Oracle Web Tier*
- *Oracle Fusion Middleware Installation Guide for Oracle WebLogic Server*
- *Oracle Fusion Middleware Installation Planning Guide*
- *Oracle Fusion Middleware Managing Oracle WebCenter Content*
- *Oracle Fusion Middleware Reference for Oracle Identity Management*
- *Oracle Fusion Middleware Repository Creation Utility User's Guide*
- *Oracle Fusion Middleware Security Guide*
- *Oracle Fusion Middleware User's Guide for Technology Adapters*
- *Oracle Fusion Middleware Using Oracle WebCenter Content*
- *Oracle Fusion Middleware Using Web Server Plug-Ins with Oracle WebLogic Server Guide*
- *Oracle WebCenter Content Administrator's Guide for Imaging*
- *Oracle WebCenter Content Developer's Guide for Imaging*
- *Oracle WebCenter Content Installation Guide*
- *Oracle WebCenter Content Quick Installation Guide*

This guide also refers to the following documents for more information:

- *Oracle Clusterware Installation Guide for Linux*
- *Oracle Coherence Developer's Guide*
- *Oracle Database Administrator's Guide*
- *Oracle Database Backup and Recovery User's Guide*
- *Oracle Real Application Clusters Administration and Deployment Guide*
- *Oracle Real Application Clusters Installation Guide for Linux and UNIX*
- *Oracle Universal Installer Concepts Guide.*
- *Understanding Domain Configuration for Oracle WebLogic Server*

Conventions

The following text conventions are used in this document.

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

1 Enterprise Deployment Overview

This chapter provides an overview of the enterprise deployment for Oracle WebCenter Content. It includes the following topics:

1.1 About the Enterprise Deployment Guide

An enterprise deployment guide 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 this blueprint spans Oracle products across the entire technology stack: Oracle Database, Oracle Fusion Middleware, Oracle Applications, and Oracle Enterprise Manager Fusion Middleware Control.

An Oracle Fusion Middleware enterprise deployment provides the following benefits:

- 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 about high availability practices, see the Oracle Maximum Availability Architecture Best Practices page on the Oracle Technology Network at `http://www.oracle.com/technetwork/database/features/availability/maa-best-practices-155366.html`.

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

1.2 About Oracle WebCenter Content

The Oracle WebCenter Content suite (formerly known as Oracle Enterprise Content Management Suite, or Oracle ECM) provides unified content management that ensures seamless access to the right information in the appropriate business context by helping organizations implement a strategic content infrastructure for managing documents, images, and rich media files while delivering contextual integration with enterprise applications through the Oracle Application Extension Framework (AXF).

Oracle WebCenter Content provides the following key benefits:

- Reduce costs: lower printing, shipping, storage, and maintenance costs
- Gain efficiencies: with a single source of truth, streamlined business processes, and more complete and faster access to information.
- Reduce risk: improve consistency and auditability, comply with business policies and regulations, ensure content security, and better manage your brand
- Create value: improve business agility and optimize revenue by improving cross-selling and up-selling; enabling your channels; and improving customer retention.

The reference enterprise deployment topology in this guide includes the following Oracle WebCenter Content feature sets:

- Oracle WebCenter Content
- Oracle WebCenter Content: Imaging
- Oracle WebCenter Content: Inbound Refinery

Oracle WebCenter Content

Oracle WebCenter Content (formerly known as Oracle Universal Content Management, or Oracle UCM) provides organizations with a unified repository to house unstructured content, and deliver it to business users in the proper format, and within the context of familiar applications to fit the way they work.

Oracle WebCenter Content: Imaging

Oracle WebCenter Content: Imaging (formerly known as Oracle Imaging and Process Management, or Oracle I/PM) is the most complete, integrated, and cost-effective imaging platform for end-to-end management of document images within enterprise business processes. It leverages Oracle WebCenter Forms Recognition for intelligent data capture and AXF for orchestration of LOB and process collaboration. It also provides annotation and markup of images, automates routing and approvals, and a scalable repository supporting enterprise-wide applications. With Imaging, organizations can quickly automate business processes in Oracle and other third-party enterprise applications.

Oracle WebCenter Content: Inbound Refinery

Oracle WebCenter Content: Inbound Refinery is a conversion server that manages file conversions for electronic assets such as documents, digital images, and motion video. In addition to conversion, Inbound Refinery provides thumbnail functionality for documents and images, storyboarding for video, and the ability to extract and use EXIF data from digital images and XMP data from electronic files generated from programs such as Adobe Photoshop and Adobe Illustrator. Organizations can use Inbound Refinery to convert content items stored in Oracle WebCenter Content Server.

1.3 Enterprise Deployment Terminology

The following terminology is used in this enterprise deployment guide:

- **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.
- **Oracle Common home**: The Oracle home that contains the binary and library files required for Oracle Enterprise Manager Fusion Middleware Control and Java Required Files (JRF).
- **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, and 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 handles the load by using the other available systems. 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 the *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, including Oracle, 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**: 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 through 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 enterprise deployment domain. Among other things, the following are located on the shared disk:
  - Middleware home software
  - AdminServer Domain Home
  - JMS
  - Tlogs (where applicable)

  Except for a WebCenter Content or Inbound Refinery Managed Server, you can locate a Managed Server home on the shared disk as well. 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 and write to.
- **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, or 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**: A 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 can 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 a UNIX system, this is the name returned by the `hostname` command.

  A 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 address of a machine on the network. In almost most 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**: A virtual host name is a network-addressable host name that maps to one or more physical machines through a load balancer or a hardware cluster. For load balancers, the term *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.

1.4 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 1.4.1, "Built-In Security"
- Section 1.4.2, "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.

1.4.1 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/technetwork/index.html`) provides a list of validated load balancers and their configuration at `http://www.oracle.com/technetwork/middleware/ias/tested-lbr-fw-sslaccel-100648.html`.

- Communication from external clients does not go beyond the Load Balancing Router (LBR) 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 across 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.
- Oracle 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.

1.4.2 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.

2

Introduction to the Enterprise Deployment Reference Topology

This chapter describes and illustrates the enterprise deployment reference topology described in this guide. The roadmap for installation and configuration directs you to the appropriate chapters for the tasks you need to perform. Use this chapter to help you plan your Oracle WebCenter Content enterprise deployment.

This chapter includes these topics:

2.1 Overview of the Enterprise Deployment Reference Topology

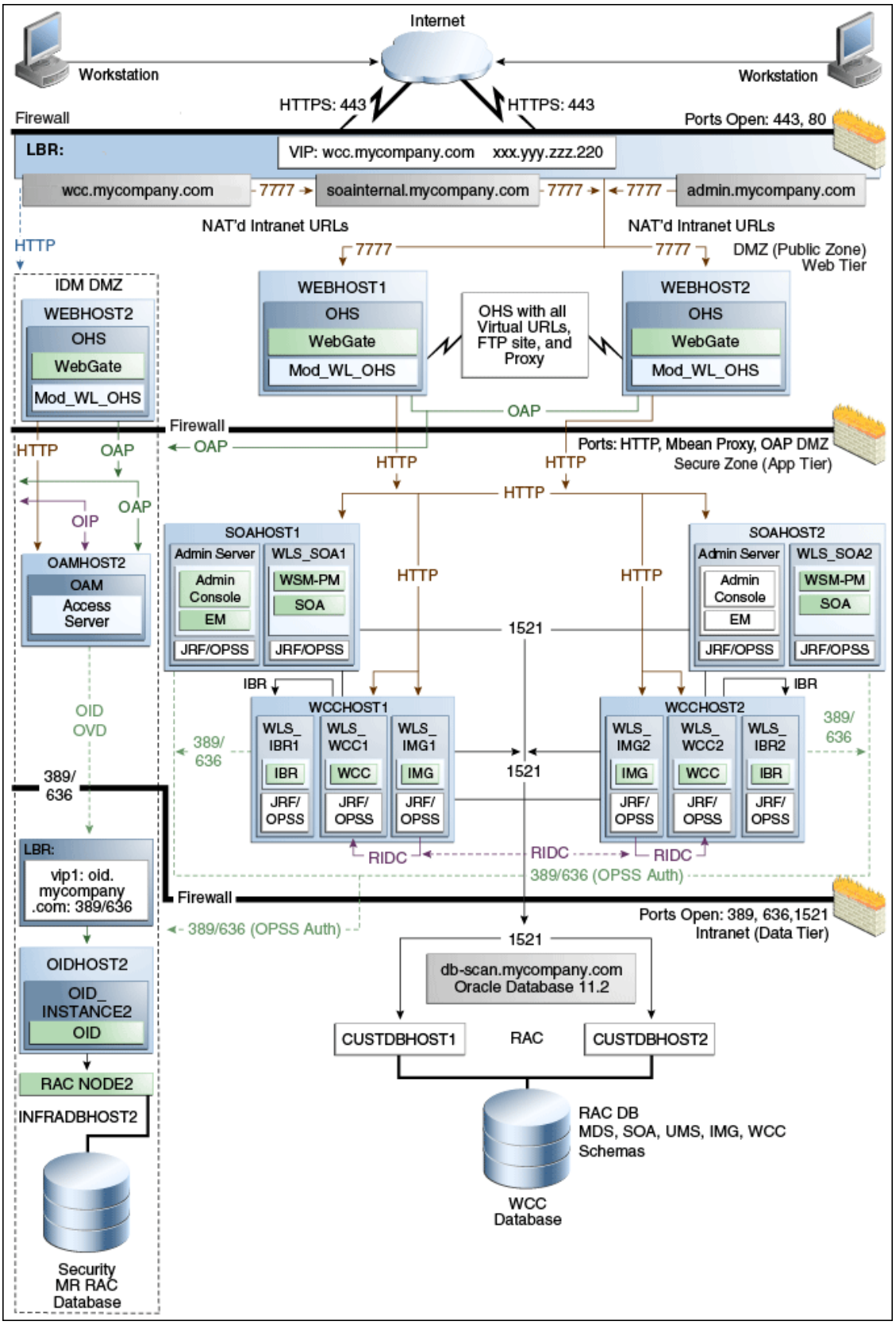
This section describes a diagram used to illustrate the enterprise deployment reference topology described in this guide. Use this section to plan your enterprise deployment topology.

This section covers these topics:

2.1.1 Reference Topology Documented in the Guide

This guide provides configuration instructions for a reference enterprise topology that uses service-oriented architecture (SOA) and Oracle WebCenter Content with Oracle Access Manager, as shown in Figure 2–1.

Figure 2–1 Reference Topology for Oracle WebCenter Content



Note: Your actual enterprise deployment topology may require variations on the topology described in this guide.

2.1.2 About Oracle Identity Management Integration

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 Oracle Platform Security Services, centralized identity and credential store, and authentication for the Oracle WebLogic Server domain. The Oracle Identity Management enterprise deployment is separate from the Oracle WebCenter Content enterprise deployment and exists in a separate domain by itself. For more information on Oracle Identity Management in an enterprise deployment context, see the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*.

The primary interface to the Oracle Identity Management enterprise deployment is the LDAP traffic to the LDAP servers, the OAP (Oracle Access Protocol) to the Oracle Access Manager Access Servers, and the HTTP redirection of authentication requests.

2.1.3 About the Web Tier Nodes

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 `*_vh.conf` files.

Through `*_vh.conf` files, which allow 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 Oracle Identity Management DMZ. WebGate and Oracle Access Manager are used to perform operations such as user authentication. 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.

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.

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

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 back-end 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 back-end 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 back-end 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 enterprise deployment.

2.1.4 About the Application Tier

Nodes in the application tier are located in the DMZ secure zone. In this tier, two nodes (SOAHOST1 and SOAHOST2) run Oracle WebLogic Server configured with Managed Servers for running Oracle SOA Suite components, such as BPEL Process Manager. The Managed Servers are configured in an active-active manner.

WCCHOST1 and WCCHOST2 run the Oracle WebCenter Content servers, Imaging and AXF servers, and Inbound Refinery servers.

SOAHOST1 and SOAHOST2 also 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 8.5.6, "Verifying Manual Failover of the Administration Server"). Alternatively, you can configure the WebLogic Server Administration Console with CFC/CRS to fail over automatically on a separate hardware cluster (not shown in this architecture).

Oracle Web Services Manager (Oracle WSM) provides a policy framework to manage and secure Web services in the enterprise deployment topology. WSM Policy Manager also runs in active-active configuration in the same servers as Oracle SOA Suite.

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.)

2.1.5 About the Data Tier

Nodes in the data tier are located in the most secured network zone (the intranet). In this tier, an Oracle Real Applications Clusters (RAC) database runs on the nodes CUSTDBHOST1 and CUSTDBHOST2. The database contains the schemas needed by the Oracle SOA Suite and Oracle WebCenter Content components. The Oracle WebCenter Content and Oracle SOA Suite 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 Oracle Identity Management enterprise deployment.

2.1.6 About the Unicast Requirement for Communication

Oracle recommends that the nodes in the Oracle WebCenter Content enterprise deployment 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.

In unicast messaging mode, the default listening port of the server is used if no channel is configured.

Cluster members communicate to the group leader when they need to send a broadcast message which is usually the heartbeat message. When the cluster members detect the failure of a group leader, the next oldest member becomes the group leader.

The frequency of communication in unicast mode is similar to the frequency of sending messages on multicast port.

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

- All members of a WebLogic Server 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 Server 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.)

2.2 Hardware Requirements for an Enterprise Deployment on Linux

Before you install and configure your enterprise deployment, review the *Oracle Fusion Middleware System Requirements and Specifications* on the Oracle Technology Network (OTN) to ensure that your environment meets the minimum installation requirements for the products you are installing.

In addition, Table 2–1 lists the typical hardware requirements for the enterprise deployment described in this guide on Linux operating systems.

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 Oracle WebCenter Content system being used.

Table 2–1 Typical Hardware Requirements

Server	Disk	Memory	TMP Directory	Swap
Database	*n*X*m* *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*	10 GB	4 GB	Default	Default
SOAHOST*n*	10 GB[1]	10 GB	Default	Default
WCCHOST*n*	10 GB[2]	10 GB	Default	Default

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

[2] Oracle WebCenter Content can reuse *MW_HOME* binaries from the Oracle SOA Suite installation in shared storage.

Notes:

- 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 Oracle SOA Suite system being used.
- For WebCenter Content, Imaging, and Inbound Refinery Managed Servers, you need to increase the size of the heap allocated for the Java Virtual Machine (JVM) on which each Managed Server runs to at least 1 GB (1024 MB). For more information, see "Increasing the Java VM Heap Size for Managed Servers" in the *Oracle WebCenter Content Installation Guide*.

2.3 Clock Synchronization

The clocks of all servers participating in the clusters must be synchronized to within one second difference to enable proper functioning of jobs and adapters. To accomplish this, use a single network time server and then point each server to that network time server.

The procedure for pointing to the network time server is different on different operating systems. Refer to your operating system documentation for more information.

2.4 Software Components to Install

Table 2–2 lists the Oracle software you will need to obtain before starting the procedures in this guide.

For complete information about downloading Oracle Fusion Middleware software, see *Oracle Fusion Middleware Download, Installation, and Configuration Readme Files* on the Oracle Technology Network (OTN).

Table 2–2 Components and Installation Sources

Component	Details
Oracle Database 10*g* or 11*g*	Oracle Database 10*g* (10.2.0.4 or later, Standard Edition or Enterprise Edition), using the AL32UTF8 character set Oracle Database 11*g* (11.1.0.7 or later, Standard Edition or Enterprise Edition), using the AL32UTF8 character set **Note:** For Oracle WebCenter Content enterprise deployments, Oracle recommends using GridLink data sources to connect to Oracle RAC databases. To use the Oracle Single Client Access Name (SCAN) feature with GridLink, the Oracle RAC database version must be Oracle Database 11gR2 (11.2 or later, Enterprise Edition).
Repository Creation Utility (RCU)	Oracle Fusion Middleware Repository Creation Utility 11*g* (11.1.1.7)
Oracle WebLogic Server	Oracle WebLogic Server (10.3.6)
Oracle HTTP Server (OHS)	Oracle Fusion Middleware WebTier and Utilities 11*g* (11.1.1.7)
Oracle SOA Suite	Oracle SOA Suite 11*g* (11.1.1.7)
Oracle WebCenter Content	Oracle WebCenter Content 11*g* (11.1.1.7)
Oracle Access Manager WebGate	WebGate 10*g* (10.1.4.3) for Oracle Access Manager 10*g* or WebGate 11*g* (11.1.1.2 or later) for Oracle Access Manager 11*g*.
Oracle Virtual Directory	Oracle Identity and Access Management 11*g* (11.1.1.5 or later)
Oracle Internet Directory	Oracle Identity and Access Management 11*g* (11.1.1.5 or later)

2.5 About LDAP As Credential and Policy Store

With Oracle Fusion Middleware, you can use different types of credential and policy stores in an Oracle WebLogic Server domain. Domains can use stores based on XML files, different types of LDAP providers, or an Oracle Database. 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 WebCenter Content enterprise deployment topology uses different domain homes for the Administration Server and the Managed Servers, as described in Section 4.3, "About Recommended Locations for the Different Directories." Derived from this, and for integrity and consistency purposes, Oracle requires the use of an LDAP server as policy and credential store in the context of an Oracle WebCenter Content enterprise deployment topology. To configure the Oracle WebCenter Content enterprise deployment topology with an LDAP as credential and policy store, follow the steps in Section 15.2, "Credential and Policy Store Configuration."

2.6 Roadmap for the Reference Topology Installation and Configuration

Before beginning your Oracle WebCenter Content enterprise deployment, review the flow chart in Figure 2–2. This flow chart illustrates the high-level process for completing the enterprise deployment documented in this guide. Table 2–3 describes the steps in the flow chart and directs you to the appropriate chapter for each step.

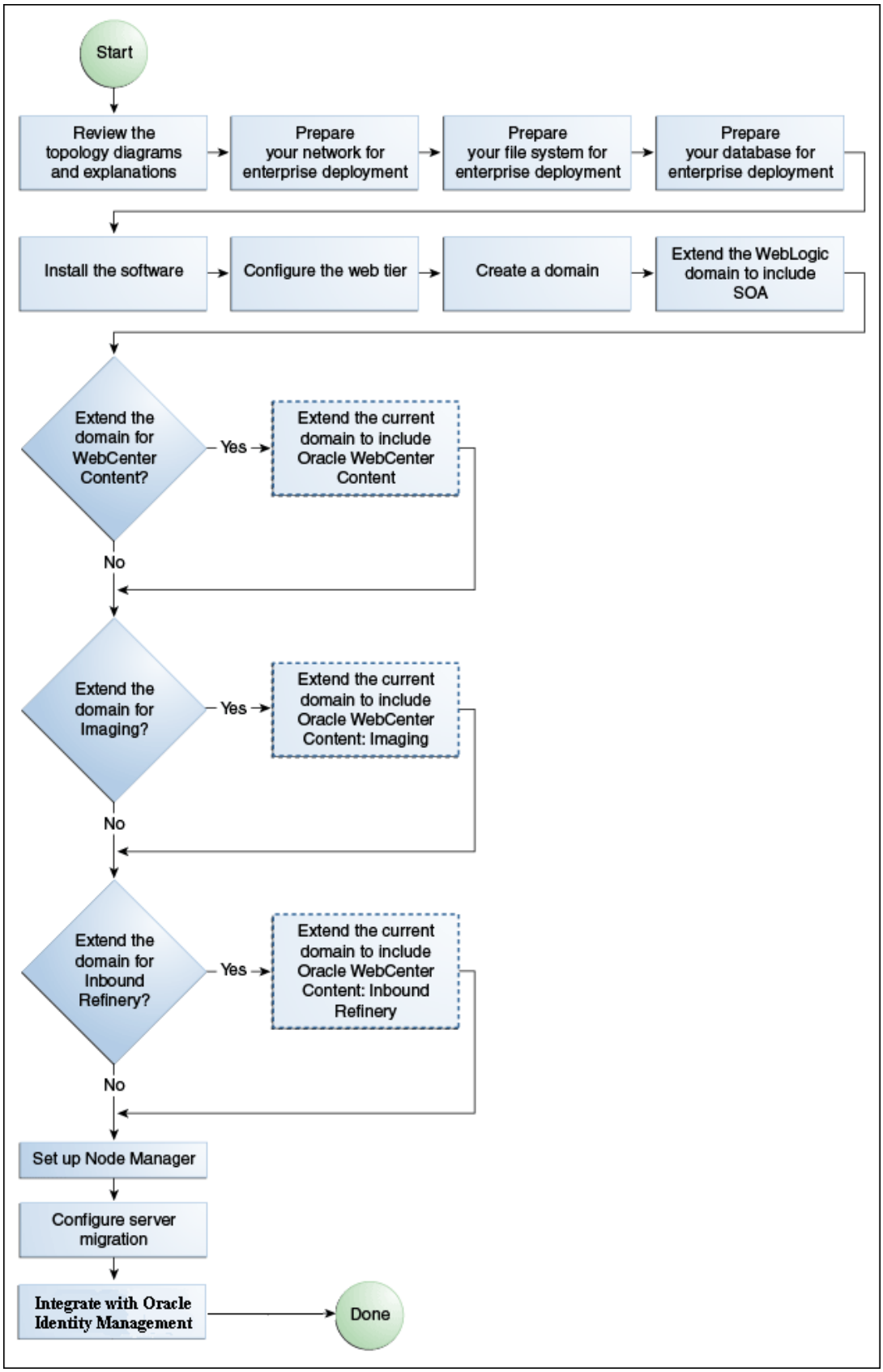
This section covers the following topics:

- Section 2.6.1, "Flow Chart of the Oracle WebCenter Content Enterprise Deployment Process"
- Section 2.6.2, "Steps in the Oracle WebCenter Content Enterprise Deployment Process"
- Section 2.6.3, "Understanding the Incremental, Modular Approach to Enterprise Deployment"

2.6.1 Flow Chart of the Oracle WebCenter Content Enterprise Deployment Process

Figure 2–2 provides a flow chart of the Oracle WebCenter Content enterprise deployment process. Review this chart to become familiar with the steps that you must follow, based on the existing environment.

Figure 2–2 Flow Chart of the Oracle WebCenter Content Enterprise Deployment Process



2.6.2 Steps in the Oracle WebCenter Content Enterprise Deployment Process

Table 2–3 describes each of the steps in the enterprise deployment process flow chart for Oracle WebCenter Content, shown in Figure 2–2. The table also provides references to more information on each step in the process.

Table 2–3 Steps in the Oracle WebCenter Content Enterprise Deployment Process

Step	Description	More Information
Prepare your network for enterprise deployment	To prepare your network for an enterprise deployment, understand concepts, such as virtual server names and IPs and virtual IPs, and configure your load balancer by defining virtual host names.	Chapter 3, "Preparing the Network for an Enterprise Deployment"
Prepare your file system for enterprise deployment	To prepare your file system for an enterprise deployment, review the terminology for directories and directory environment variables, and configure shared storage.	Chapter 4, "Preparing the File System for an Enterprise Deployment"
Prepare your database for enterprise deployment	To prepare your database for an enterprise deployment, review database requirements, create database services, load the metadata repository in the Oracle RAC database, configure SOA, WCC and IMG schemas for transactional recovery privileges, and back up the database.	Chapter 5, "Preparing the Database for an Enterprise Deployment"
Install the software	Install Oracle HTTP Server, Oracle WebLogic Server, and Oracle Fusion Middleware, and apply patchsets to Oracle Fusion Middleware components.	Chapter 6, "Installing the Software for an Enterprise Deployment"
Configure the web tier	Configure Oracle HTTP Server with the load balancer, and configure virtual host names.	Chapter 7, "Configuring Oracle Web Tier"
Create a domain	Run the Oracle Fusion Middleware Configuration Wizard to create a domain.	Chapter 8, "Creating a Domain for an Enterprise Deployment"
Extend the domain for Oracle SOA Suite	Extend the existing WebLogic Server domain by running the Fusion Middleware Configuration Wizard and configuring Oracle SOA Suite components.	Chapter 9, "Extending the Domain to Include Oracle SOA Suite Components"
Extend the domain for Oracle WebCenter Content	Extend the existing WebLogic Server domain by running the Fusion Middleware Configuration Wizard and configuring Oracle WebCenter Content.	Chapter 10, "Extending the Domain to Include Oracle WebCenter Content"
Extend the domain for Oracle WebCenter Content: Imaging	Extend the existing WebLogic Server domain by running the Fusion Middleware Configuration Wizard and configuring Oracle WebCenter Content: Imaging.	Chapter 11, "Extending the Domain to Include Imaging"

Table 2–3 (Cont.) Steps in the Oracle WebCenter Content Enterprise Deployment Process

Step	Description	More Information
Extend the domain for Oracle WebCenter Content: Inbound Refinery	Extend the existing WebLogic Server domain by running the Fusion Middleware Configuration Wizard and configuring Oracle WebCenter Content: Inbound Refinery.	Section 12, "Extending the Domain to Include Inbound Refinery"
Set up Node Manager	Set up Node Manager by enabling host name verification, starting Node Manager, and configuring WebLogic Server domains to use custom keystores.	Section 13, "Setting Up Node Manager"
Configure Server Migration	Configure server migration for the WLS_SOA*n* and WLS_IMG*n* Managed Servers. The WLS_SOA1 and WLS_IMG1 Managed Servers are configured to restart on SOAHOST2 and WCCHOST2, respectively, should a failure occur. The WLS_SOA2 and WLS_IMG2 Managed Servers are configured to restart on SOAHOST1 and WCCHOST1, respectively, should a failure occur.	Section 14, "Configuring Server Migration for an Enterprise Deployment"
Oracle Identity Management integration	You can integrate your Oracle WebCenter Content enterprise deployment with Oracle Access Manager 10*g* or 11*g*.	Section 15, "Integrating with Oracle Identity Management"

2.6.3 Understanding the Incremental, Modular Approach to Enterprise Deployment

By design, this document describes an incremental and modular approach to setting up an enterprise deployment.

The instructions for setting up the storage, database, networking, and Web Tier infrastructure are similar to the instructions provided in the other Oracle Fusion Middleware Enterprise Deployment Guides. These elements of the topology provide the foundation for the Oracle WebLogic Server domain you later configure to support the enterprise deployment.

When you create the domain, the instructions vary from guide to guide. However, all the Enterprise Deployment Guides provide separate, modular instructions for creating and extending an Oracle WebLogic Server domain, as follows:

1. Install the Oracle Fusion Middleware software on disk and create the necessary binary directories.
2. Run the Fusion Middleware Configuration Wizard to create the domain and configure only the administration components.

   The administration components include the Administration Server, Oracle WebLogic Server Administration Console, Oracle Enterprise Manager Fusion Middleware Control, and Oracle Web Services Manager.
3. Run the Fusion Middleware Configuration Wizard again to extend the domain to include the primary Oracle Fusion Middleware product you want to use.
4. Optionally, run the Fusion Middleware Configuration Wizard again to extend the domain to include other supporting components and products.

This incremental approach allows you to verify the environment after each pass of the Fusion Middleware Configuration Wizard. It also simplifies troubleshooting during the setup process.

In addition, this modular approach allows you to consider alternative topologies. Specifically, after you configure the Administration components, the domain you create does not need to contain all the components described in this guide. Instead, you can use the domain extension chapters independently and selectively, to configure individual components that are required for your specific organization.

3

Preparing the Network for an Enterprise Deployment

This chapter describes the network environment preconfiguration required by the Oracle WebCenter Content enterprise deployment topology. Use this chapter to plan your configuration of virtual server names, load balancers, IPs and Virtual IPs, and firewalls and ports.

This chapter includes the following topics:

3.1 Overview of Preparing the Network for an Enterprise Deployment

You must configure several virtual servers and associated ports on the load balancer for different types of network traffic and monitoring. These virtual servers should be configured to the appropriate real hosts and ports for the services running. Also, the load balancer should be configured to monitor the real host and ports for availability so that the traffic to these is stopped as soon as possible when a service is down. This ensures that incoming traffic on a given virtual host is not directed to an unavailable service in the other tiers.

3.2 Virtual Server Names Used by the Topology

The Oracle WebCenter Content enterprise topology uses the following virtual server names:

- wcc.mycompany.com
- admin.mycompany.com
- soainternal.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.

You will define the virtual server names on the load balancer using the procedure in Section 3.3, "Configuring the Load Balancers."

3.2.1 wcc.mycompany.com

`wcc.mycompany.com` is a virtual server name that acts as the access point for all HTTP traffic to the run-time Oracle WebCenter Content components. Traffic to SSL is configured. Clients access this service using the address `wcc.mycompany.com:443`.

3.2.2 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 Oracle WebLogic Server Administration Console and Oracle Enterprise Manager Fusion Middleware Control.

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.

3.2.3 soainternal.mycompany.com

`soainternal.mycompany.com` is a virtual server name used for internal invocations of SOA and WebCenter Content services. This URL is not exposed to the Internet and is accessible only from the intranet. (For Oracle SOA Suite systems, users can set this URL while modeling composites or at runtime with the appropriate MBean through Fusion Middleware Control, as the URL to be used for invoking internal services.) You can use this URL for any service call back to WebCenter Content components from Oracle SOA Suite.

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.

3.3 Configuring the Load Balancers

Several virtual servers and associated ports must be configured on the load balancer for different types of network traffic and monitoring. These should be configured to the appropriate real hosts and ports for the services running. Also, the load balancer should be configured to monitor the real host and ports for availability so that the traffic to these is stopped as soon as possible when a service is down. This ensures that incoming traffic on a given virtual host is not directed to an unavailable service in the other tiers.

There are two load balancer devices in the recommended topologies. One load balancer is set up for external HTTP traffic and the other load balancer is set up for internal LDAP traffic. A deployment may choose to have a single load balancer device due to a variety of reasons. While this is supported, the deployment should consider the security implications of doing this and if found appropriate, open up the relevant firewall ports to allow traffic across the various DMZs. In either case, it is highly recommended to deploy a given load balancer device in fault-tolerant mode.

This enterprise topology uses an external load balancer. Configure the load balancer by defining the virtual server names described in Section 3.2, "Virtual Server Names Used by the Topology."

For more information on load balancers, see Section 2.1.3, "About the Web Tier Nodes."

3.3.1 Load Balancer Requirements

The enterprise topologies use an external load balancer. This external load balancer must 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.
- Monitoring of ports (HTTP and HTTPS).
- 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 WebLogic Server clusters, the load balancer must 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.
- Resource monitoring, port monitoring, and process failure detection: The load balancer must be able to detect service and node failures (through notification or some other means) and to stop directing non-Oracle Net traffic to the failed node. If your external load balancer has the ability to automatically detect failures, you should use it.
- Fault-tolerant mode: It is highly recommended that you configure the load balancer to be in fault-tolerant mode.
- Other: It is highly recommended that you configure the load-balancer virtual server to return immediately to the calling client when the back-end 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.
- SSL acceleration: This feature is required.
- Configure the virtual server or servers in the load balancer for the directory tier with a high value for the connection timeout for TCP connections. This value should be more than the maximum expected time over which no traffic is expected between the Oracle Access Manager and the directory tier.
- Ability to preserve the client IP addresses: The load balancer must have the capability to insert the original client IP address of a request in an X-Forwarded-For HTTP header to preserve the client IP address.

The Oracle Technology Network provides a list of validated load balancers and their configuration at http://www.oracle.com/technetwork/middleware/ias/tested-lbr-fw-sslaccel-100648.html.

3.3.2 Load Balancer Configuration Procedures

The procedures for configuring a load balancer differ, depending on the specific type of load balancer. Refer to the vendor supplied documentation for actual steps. The following steps outline the general configuration flow:

1. Create a pool of servers. This pool contains a list of servers and the ports that are included in the load balancing definition. For example, for load balancing between the web hosts, you create a pool of servers that would direct requests to hosts WEBHOST1 and WEBHOST2 on port 7777.
2. Create rules to determine whether or not a given host and service is available and assign it to the pool of servers described in Step 1.
3. Configure a virtual server in the load balancer for `soainternal.mycompany.com:80`, and define the following rules for this virtual server:
   - Use your internal SOA address and internal Oracle WebCenter Content 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, nodes, or both 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 `wcc.mycompany.com:443`, and define the following rules for this virtual server:
   - Use your system's front-end address as the virtual server address (for example, `wcc.mycompany.com`). The front-end address is the externally facing host name that is used by your system and 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 HTTP as the protocol.
   - Enable address and port translation.
   - Enable reset of connections when services, nodes, or both 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.
5. Configure a virtual server in the load balancer for `admin.mycompany.com:80`, and define the following rules 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, nodes, or both are down.

- Optionally, create rules to allow access only to `/console` and `/em` on this virtual server.
- Assign the pool created in step 1 to the virtual server.

6. 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 /\n\n` 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 response time that you can expect from your Oracle WebCenter Content system, that is, specify a value greater than the longest period of time any of your requests to HTTP servers can take.

After you configure the virtual hosts in Section 7.5, "Defining Virtual Hosts," you should be able to access the virtual host name addresses. If you cannot access them, review this procedure to ensure that it was completed correctly.

3.4 Configuring 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 3–1. As shown in this figure, each VIP and IP is attached to the Oracle WebLogic Server instance 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 the Oracle WebLogic Server migration feature. WLS_IMG1 and WLS_IMG2 also use server migration to fail over VIP4 and VIP5, respectively, from WCCHOST1 to WCCHOST2.

For information about the WebLogic Server migration feature, see the *Oracle Fusion Middleware High Availability Guide*.

Physical IPs (non-virtual) are fixed to each node. IP1 is the physical IP of WCCHOST1 and is used as the listen address by the WLS_WCC1 server. IP2 is the physical IP of WCCHOST2 and is used as the listen address by the WLS_WCC2 server.

The virtual IP `db-scan.mycompany.com` is the database Oracle Single Client Access Name (SCAN) address for a GridLink data source.

Figure 3–1 IPs and VIPs Mapped to Administration Server and Managed Servers

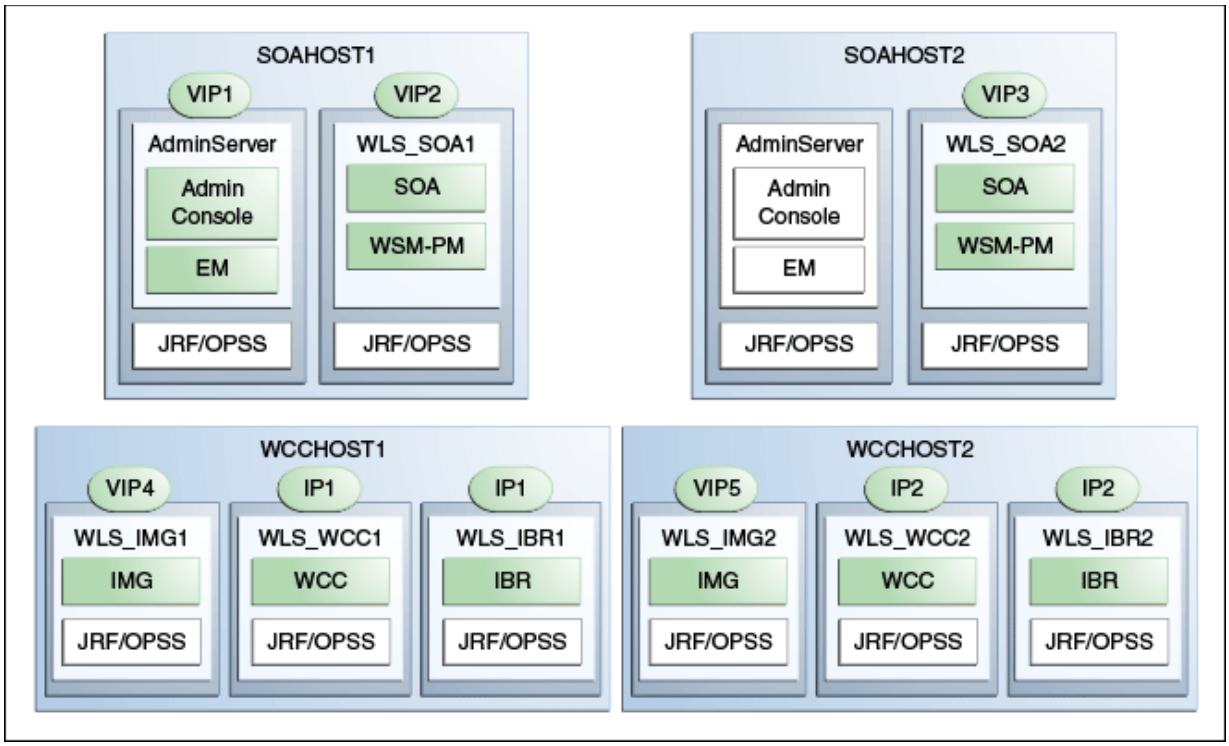


Table 3–1 provides descriptions of the various virtual hosts.

Table 3–1 Virtual Hosts

Virtual IP	VIP Maps to...	Description
VIP1	ADMINVHN	ADMINVHN 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	SOAHOST1VHN1	SOAHOST1VHN1 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).
VIP4	WCCHOST1VHN1	WCCHOST1VHN1 is the virtual host name that maps to the listen address for WLS_IMG1 and fails over with server migration of this Managed Server. It is enabled on the node where WLS_IMG1 process is running (WCCHOST1 by default).
VIP5	WCCHOST2VHN1	WCCHOST2VHN1 is the virtual host name that maps to the listen address for WLS_IMG2 and fails over with server migration of this Managed Server. It is enabled on the node where WLS_IMG2 process is running (WCCHOST2 by default).
db-scan.mycompany.com		`db-scan.mycompany.com` is the database Oracle Single Client Access Name (SCAN) address for a GridLink data source.

3.5 Enabling Virtual IP Addresses for an Enterprise Deployment

This step is required for failover of the WebLogic Server Administration Server, regardless of whether other Oracle Fusion Middleware components are installed later or not.

You associate an Administration Server with a virtual IP address. This allows the Administration Server to be started on a different host if the primary host fails.

To enable a virtual IP address from a Linux system, run the following commands as `root`:

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

where *`interface`* is `eth0` or `eth1`, and *`index`* is `0`, `1`, or `2`.

For example:

```
/sbin/ifconfig eth0:1 100.200.140.206 netmask 255.255.255.0
```

Enable your network to register the new location of the virtual IP address:

```
/sbin/arping -q -U -c 3 -I eth0 100.200.140.206
```

Validate that the address is available by issuing the `ping` command from another node; for example:

```
/bin/ping 100.200.140.206
```

3.6 Configuring 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 ensure that the same port number is not used by two services on a host.

Most port numbers are assigned during installation.

Table 3–2 lists the ports used in the Oracle WebCenter Content 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 3–2 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 Oracle SOA Suite.
Browser request	FW0	443	HTTPS / Load Balancer	Inbound	Timeout depends on all HTML content and the type of process model used for Oracle SOA Suite.
Browser request	FW1	80	HTTP / Load Balancer	Outbound (for intranet clients)	Timeout depends on all HTML content and the type of process model used for Oracle SOA Suite.
Browser request	FW1	443	HTTPS / Load Balancer	Outbound (for intranet clients)	Timeout depends on all HTML content and the type of process model used for Oracle SOA Suite.
Callbacks and outbound invocations	FW1	80	HTTP / Load Balancer	Outbound	Timeout depends on all HTML content and the type of process model used for Oracle SOA Suite.
Callbacks and Outbound invocations	FW1	443	HTTPS / Load Balancer	Outbound	Timeout depends on all HTML content and the type of process model used for Oracle SOA Suite.
Oracle Notification Server (ONS)	FW2	6200	ONS	Both	Both required for Gridlink. An ONS server runs on each database server.
Load balancer to Oracle HTTP Server	n/a	7777	HTTP	n/a	See Section 3.3, "Configuring the Load Balancers."
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).
Oracle SOA Suite and Oracle WSM server access	FW1	8001 Range: 8000 - 8080	HTTP / WLS_SOA*n*	Inbound and Outbound	Timeout varies based on the type of process model used for Oracle SOA Suite.
Oracle WebCenter Content access	FW1	16200	HTTP / WLS_WCC*n*	Inbound	Browser-based access. Configurable session timeouts.
Oracle WebCenter Content: Imaging access	FW1	16000	HTTP / WLS_IMG*n*	Inbound	Browser-based access. Configurable session timeouts.

Table 3–2 (Cont.) Ports Used

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Imaging connection to Oracle WebCenter Content	n/a	4444	TCP/IP Unicast	n/a	Persistent connection. Timeout configurable on Content Server.
Communication between SOA_Cluster members	n/a	8001	TCP/IP Unicast	n/a	By default, this communication uses the same port as the server's listen address.
Communication between WCC_Cluster members	n/a	16200	TCP/IP Unicast	n/a	By default, this communication uses the same port as the server's listen address.
Communication between IMG_Cluster members	n/a	16000	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 t3	HTTP / Administration Server and Oracle Enterprise Manager Fusion Middleware Control	Both	You should tune this timeout based on the type of access to the administration console (whether it is planned to use the 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 "About Firewalls and Ports" in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*.
Access Server access	F	5575 (Oracle Access Manager 11*g*) 6021 (Oracle Access Manager 10*g*)	OAP	Inbound	For actual values, see "About Firewalls and Ports" in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*.
Identity Server access (Oracle Access Manager 10*g*)	FW1	6022	OAP	Inbound	n/a

Table 3–2 (Cont.) Ports Used

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Database access	FW2	1521	SQL*Net	Both	Timeout depends on all database content and on the type of process model used for Oracle SOA Suite.
Coherence for deployment	n/a	8088 Range: 8000 - 8090		n/a	n/a
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.
JOC for OWSM	n/a	9991 Range: 9988-9998	TCP/IP	n/a	n/a
Browser request	n/a	16250 16250	HTTP / WLS_IBR*n*	n/a	Browser-based access. Configurable session timeouts.

Note: The firewall ports depend on the definition of TCP/IP ports.

4

Preparing the File System for an Enterprise Deployment

This chapter describes how to prepare the file system for an Oracle WebCenter Content enterprise deployment. It provides information about recommended directory structure and locations, and includes a procedure for configuring shared storage.

This chapter includes the following sections:

4.1 Overview of Preparing the File System for Enterprise Deployment

It is important to set up your file system in a way that makes the enterprise deployment easier to understand, configure, and manage. Oracle recommends setting up your file system according to information in this chapter. The terminology defined in this chapter is used in diagrams and procedures throughout this guide.

Use this chapter as a reference to help understand the directory variables used in the installation and configuration procedures. 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.

4.2 Terminology for Directories and Directory Environment Variables

This section describes the directory environment variables used throughout this guide for configuring the Oracle WebCenter Content enterprise deployment. The following directory variables are used to describe the directories installed and configured in the guide:

- **ORACLE_BASE:** This environment variable and related directory path refer to the base directory under which Oracle products are installed.
- **MW_HOME:** This environment variable and related directory path refer to the location where Oracle Fusion Middleware resides.
- **WL_HOME:** This environment variable and related directory path contain installed files necessary to host an Oracle WebLogic Server.

- **ORACLE_HOME:** This environment variable and related directory path refer to the location where Oracle SOA Suite or Oracle WebCenter Content is installed.
- **ORACLE_COMMON_HOME:** This environment variable and related directory path refer to the Oracle home that contains the binary and library files required for Oracle Enterprise Manager Fusion Middleware Control and Java Required Files (JRF).
- **DOMAIN Directory:** This directory path refers to the location where the WebLogic Server domain information (configuration artifacts) is stored. Different Oracle WebLogic Servers can use different domain directories even when in the same node, as described in the following text.
- **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.
- **JAVA_HOME:** This is the location where JRockit is installed.
- **ASERVER_HOME:** This is the primary location of the domain configuration.
- **MSERVER_HOME:** This is a copy of the domain configuration used to start and stop Managed Servers.
- **WEBGATE_ORACLE_HOME:** This is the location of the WebGate installation.

Tip: You can simplify directory navigation by using environment variables as shortcuts to the locations in this section. For example, you could use an environment variable called `$ORACLE_BASE` to refer to `/u01/app/oracle/` (that is, the recommended *`ORACLE_BASE`* location).

4.3 About Recommended Locations for the Different Directories

The following sections describe some basic recommendations for using shared storage for an enterprise deployment topology:

- Section 4.3.1, "Shared Storage Recommendations for Binary (Oracle Home) Directories"
- Section 4.3.2, "Shared Storage Recommendations for Domain Configuration Files"
- Section 4.3.3, "Shared Storage Recommendations for JMS File Stores and Transaction Logs"
- Section 4.3.4, "Recommended Directory Locations"
- Section 4.3.5, "Directory Structure and Configurations"

4.3.1 Shared Storage Recommendations for Binary (Oracle Home) Directories

The following sections describe guidelines for using shared storage for your Oracle Fusion Middleware Oracle home directories:

- Section 4.3.1.1, "About the Binary (Oracle Home) Directories"
- Section 4.3.1.2, "About Sharing a Single Oracle Home for Multiple Domains"
- Section 4.3.1.3, "About Using Redundant Binary (Oracle Home) Directories"

4.3.1.1 About the Binary (Oracle Home) Directories

When you install any Oracle Fusion Middleware product, you install the product binaries into an Oracle home. The binary files installed in the Oracle home are read-only and remain unchanged unless the Oracle home is patched or upgraded to a newer version.

In a typical production environment, the Oracle home files are saved in a separate location from the domain configuration files, which you create using the Oracle Fusion Middleware Configuration Wizard.

The Middleware home for an Oracle Fusion Middleware installation contains the binaries for Oracle WebLogic Server, the Oracle Fusion Middleware infrastructure files, and any Oracle Fusion Middleware product-specific directories.

For more information about the structure and contents of an Oracle Fusion Middleware Oracle home, see *Oracle Fusion Middleware Concepts*.

4.3.1.2 About Sharing a Single Oracle Home for Multiple Domains

Oracle Fusion Middleware enables you to configure multiple Oracle WebLogic Server domains from a single Oracle home. This allows you to install the Oracle home in a single location on a shared volume and reuse the Oracle home for multiple hosts installations.

> **Note:** The domain directories of the Managed Servers can reside in local or shared storage, except that the WebCenter Content and Inbound Refinery Managed Servers cannot reside in shared storage. These Managed Servers include Oracle WebCenter Content Server, which uses node-specific files, such as `intradoc.cfg`.

When an Oracle home is shared by multiple servers on different hosts, there are some best practices to keep in mind. In particular, be sure that the Oracle Inventory (oraInventory) on each host is updated for consistency and for the application of patches.

To update the oraInventory for a host and attach an Oracle home on shared storage, use the following command:

```
ORACLE_HOME/oui/bin/attachHome.sh
```

For more information about the oraInventory, see "Oracle Universal Installer Inventory" in the *Oracle Universal Installer Concepts Guide.*

4.3.1.3 About Using Redundant Binary (Oracle Home) Directories

For maximum availability, Oracle recommends using redundant binary installations on shared storage.

In this model, you install two identical Oracle homes for your Oracle Fusion Middleware software on two different shared volumes. You then mount one of the Oracle homes to one set of servers, and the other Oracle home to the remaining servers. Each Oracle home has the same mount point, so the Oracle home always has the same path, regardless of which Oracle home the server is using.

Should one Oracle home become corrupted or unavailable, only half your servers are affected. For additional protection, Oracle recommends that you disk mirror these volumes.

If separate volumes are not available on shared storage, Oracle recommends simulating separate volumes using different directories within the same volume and mounting these to the same mount location on the host side. Although this does not guarantee the protection that multiple volumes provide, it does allow protection from user deletions and individual file corruption.

4.3.2 Shared Storage Recommendations for Domain Configuration Files

The following sections describe guidelines for using shared storage for the Oracle WebLogic Server domain configuration files you create when you configure your Oracle Fusion Middleware products in an enterprise deployment:

- Section 4.3.2.1, "About Oracle WebLogic Server Administration and Managed Server Domain Configuration Files"
- Section 4.3.2.2, "Shared Storage Requirements for Administration and Managed Server Domain Configuration Files"

4.3.2.1 About Oracle WebLogic Server Administration and Managed Server Domain Configuration Files

When you configure an Oracle Fusion Middleware product, you create or extend an Oracle WebLogic Server domain. Each Oracle WebLogic Server domain consists of a single Administration Server and one or more Managed Servers.

For more information about Oracle WebLogic Server domains, see *Understanding Domain Configuration for Oracle WebLogic Server*.

In an enterprise deployment, it is important to understand that the Managed Servers in a domain can be configured for active-active high availability. However, the Administration server cannot. The Administration Server is a singleton service. That is, it can be active on only one host at any given time.

4.3.2.2 Shared Storage Requirements for Administration and Managed Server Domain Configuration Files

Oracle recommends creating two copies of the domain configuration files:

- One copy is for the Administration Server configuration files.

  You install this directory on shared storage and mount it exclusively to the host that is running the Administration Server.

  In the event of the failure of that host, you can mount the directory on a different host, and the administration server started on that host.

- The other copy is for the Managed Server configuration files.

  The domain directories of the Managed Servers can reside in local or shared storage, except that the WebCenter Content and Inbound Refinery Managed Servers cannot reside in shared storage. These Managed Servers include Oracle WebCenter Content Server, which uses node-specific files, such as `intradoc.cfg`.

  Sharing domain directories for Managed Servers facilitates the scale-out procedures. However, sharing the Managed Server configuration files can also have a potential performance impact.

  As a result, the deployment you decide upon should conform to the requirements (if any) of the storage system. Some storage systems offer configuration options to facilitate multiple machines mounting the same shared volume.

The configuration steps provided for this enterprise deployment topology assume that a local domain directory for each node is used for each Managed Server.

4.3.3 Shared Storage Recommendations for JMS File Stores and Transaction Logs

JMS file stores and JTA transaction logs must be placed on shared storage to ensure that they are available from multiple hosts for recovery in the case of a server failure or migration.

For more information about saving JMS and JTA information in a file store, see "Using the WebLogic Persistent Store" in *Administering Server Environments for Oracle WebLogic Server*.

4.3.4 Recommended Directory Locations

Based on the previous assumptions, the following paragraphs describe the recommended directories. 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. 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`)
- Mounted from: Nodes alternatively mount `VOL1` or `VOL2` in such a way that at least half of the nodes use an installation and the other half use the other one. In the Oracle WebCenter Content enterprise deployment, SOAHOST1 and WCCHOST1 mount `VOL1` and SOAHOST2 and WCCHOST2 mount `VOL2`. When only one volume is available, nodes mount two different directories in shared storage alternatively (for example, SOAHOST1 would use *`ORACLE_BASE`*`/product/fmw1/` as a shared storage location, and SOAHOST2 would use *`ORACLE_BASE`*`/product/fmw2/` as a shared storage location).

Note: When there is just one volume available in the shared storage, you can provide redundancy using different directories to protect from accidental file deletions and for patching purposes. Two Middleware homes would be available; at least one at *`ORACLE_BASE`*`/product/fmw1/` and another at *`ORACLE_BASE`*`/product/fmw2/`. These Middleware homes are mounted on the same mount point in all nodes.

ORACLE_HOME (web tier):

ORACLE_BASE/product/fmw/web/

- Mount point: *ORACLE_BASE*/product/fmw/
- Shared storage location: *ORACLE_BASE*/product/fmw/ (VOL1 and VOL2)
- Mounted from: For shared storage installations, nodes alternatively mount VOL1 or VOL2 in such a way that at least half of the nodes use an installation and the other half use the other one. In the WebCenter Content enterprise deployment, WEBHOST1 would mount VOL1, and WEBHOST2 would mount VOL2. When only one volume is available, nodes mount the two suggested directories in shared storage alternatively (that is, WEBHOST1 would use *ORACLE_BASE*/product/fmw1/ as a shared storage location, and WEBHOST2 would use *ORACLE_BASE*/product/fmw2/ as a shared storage location).

> **Note:** Web tier installation is usually performed on storage local to the WEBHOST nodes. When using shared storage, appropriate security restrictions for access to the storage device across tiers need to be considered.

WL_HOME:

MW_HOME/wlserver_10.3/

ORACLE_HOME:

MW_HOME/soa/ or *MW_HOME*/wcc/

ORACLE_COMMON_HOME:

MW_HOME/oracle_common/

ORACLE_INSTANCE (OHS 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* directory is added by the Fusion Middleware Configuration Wizard.)

- Mount point on machine: *ORACLE_BASE*/admin/*domain_name*/aserver/
- Shared storage location: *ORACLE_BASE*/admin/*domain_name*/aserver/
- Mounted from: Only the node where the Administration Server is running needs to mount this directory. When the Administration Server is relocated (failed over) to a different node, the node then mounts the same shared storage location on the same mount point. The remaining nodes in the topology do not need to mount this location.

Domain directory for Managed Server 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 dependent on shared storage. The preceding example is specific to NAS, but other storage types may provide this redundancy with different types of mappings.

Location of JMS file-based stores and tlogs:

ORACLE_BASE/admin/*domain_name*/*cluster_name*/jms/

ORACLE_BASE/admin/*domain_name*/*cluster_name*/tlogs/

- Mount point: *ORACLE_BASE*/admin/*domain_name*/*cluster_name*/
- Shared storage location: *ORACLE_BASE*/admin/*domain_name*/*cluster_name*/
- Mounted from: All nodes running Oracle SOA Suite and Oracle WebCenter Content components need to mount this shared storage location so that transaction logs and JMS stores are available when server migration to another node take place.

> **Note:** The jms and tlogs directories are created for soa_cluster and img_cluster.

Location of Oracle WebCenter Content: Imaging input files, images, samples, and Viewer Cache directories:

ORACLE_BASE/admin/*domain_name*/*img_cluster_name*/input_files/

ORACLE_BASE/admin/*domain_name*/*img_cluster_name*/input_files/Samples/

ORACLE_BASE/admin/*domain_name*/*img_cluster_name*/images/

ORACLE_BASE/admin/*domain_name*/*img_cluster_name*/ViewerCache/

- Mount point: *ORACLE_BASE*/admin/*domain_name*/*img_cluster_name*/
- Shared storage location: *ORACLE_BASE*/admin/*domain_name*/*img_cluster_name*/
- Mounted from: All nodes containing Imaging mount these locations (all nodes need to have access to input files and the images to process).

The location of input files and images may vary according to each customer's implementation needs. It is relevant, however, that image files are located in a device isolated from other concurrent accesses that can degrade the performance of the system. A separate volume can be used for this purpose. In general, it is good practice to place the files under the cluster directory structure for consistent backups and maintenance.

In a multinode Imaging installation, this location is shared among all the input agents and must be accessible by all agents. If input agents are on different machines, this must be a shared network.

Note: To process input files, the input agent must have the appropriate permissions on the input directory, which must allow file locking. The input agent requires that the user account that is running the WebLogic Server service have read and write privileges to the input directory and all files and subdirectories in the input directory. These privileges are required so that the input agent can move the files to the various directories as it works on them. File locking on the share is needed by the input agent to coordinate actions between servers in the cluster.

Location of the Oracle WebCenter Content vault (native file repository):

ORACLE_BASE/admin/*domain_name*/*wcc_cluster_name*/cs/vault/

- Mount point: *ORACLE_BASE*/admin/*domain_name*/*wcc_cluster_name*/
- Shared storage location: *ORACLE_BASE*/admin/*domain_name*/*wcc_cluster_name*/
- Mounted from: All nodes that contain WebCenter Content mount this location (all nodes need to have access to input files and the images to process).

Location of the Inbound Refinery directory:

ORACLE_BASE/admin/*domain_name*/*ibr_cluster_name*/

ORACLE_BASE/admin/domain_name/*ibr_cluster_name*/ibr*N*/ (where *N* is the number of an Inbound Refinery instance)

- Mount point: *ORACLE_BASE*/admin/*domain_name*/*ibr_cluster_name*/
- Shared storage location: *ORACLE_BASE*/admin/*domain_name*/*ibr_cluster_name*/
- Mounted from: All nodes that contain Inbound Refinery mount this location.

Location of the applications directory for the Administration Server:

ORACLE_BASE/admin/*domain_name*/aserver/applications/

- Mount point: *ORACLE_BASE*/admin/*domain_name*/aserver/applications/
- Shared storage location: *ORACLE_BASE*/admin/*domain_name*/aserver/

Location of the applications directory for a Managed Server:

ORACLE_BASE/admin/*domain_name*/mserver/applications/

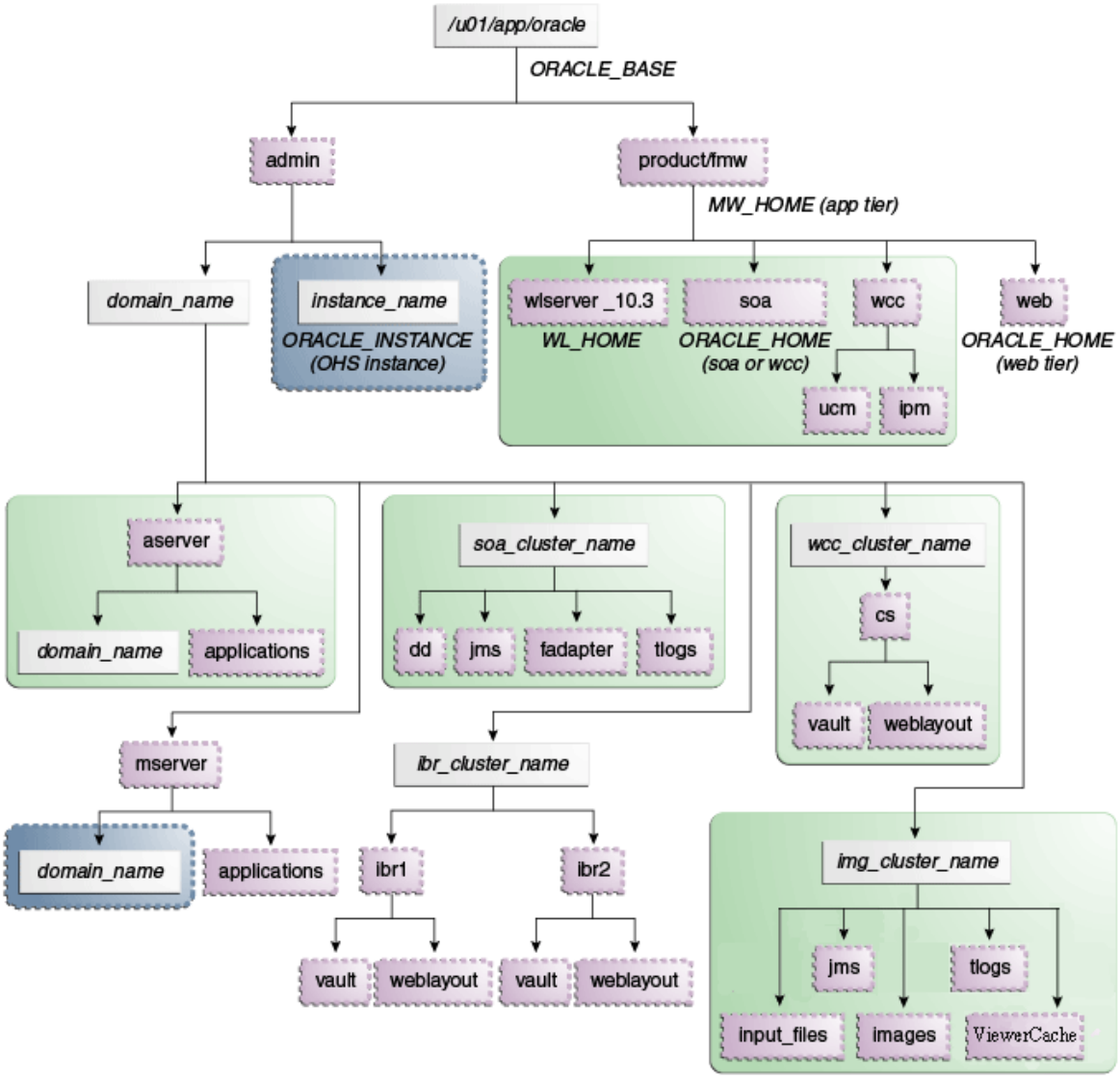
Note: This directory is local in the context of the Oracle WebCenter Content enterprise deployment. A shared domain directory for a Managed Server with Content Server does not work because certain files within the domain, such as `intradoc.cfg`, are specific to each node.

4.3.5 Directory Structure and Configurations

This section provides diagrams and to help illustrate the recommended directory structure and shared storage.

Figure 4–1 shows this directory structure in a diagram.

Figure 4–1 Enterprise Deployment Directory Structure for Oracle WebCenter Content



The directory structure in Figure 4–1 does not show other required internal directories, such as `oracle_common` and `jrockit`.

Table 4–1 explains what the various color-coded elements in Figure 4–1 mean.

Table 4–1 Directory Structure Elements

Element	Explanation
	The Administration Server domain directories, applications, deployment plans, file adapter control directory, JMS and TX logs, and the entire Middleware home are on a shared disk.
	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 4–2 shows an example configuration for shared storage with multiple volumes for Oracle SOA Suite.

Figure 4–2 Example Configuration for Shared Storage

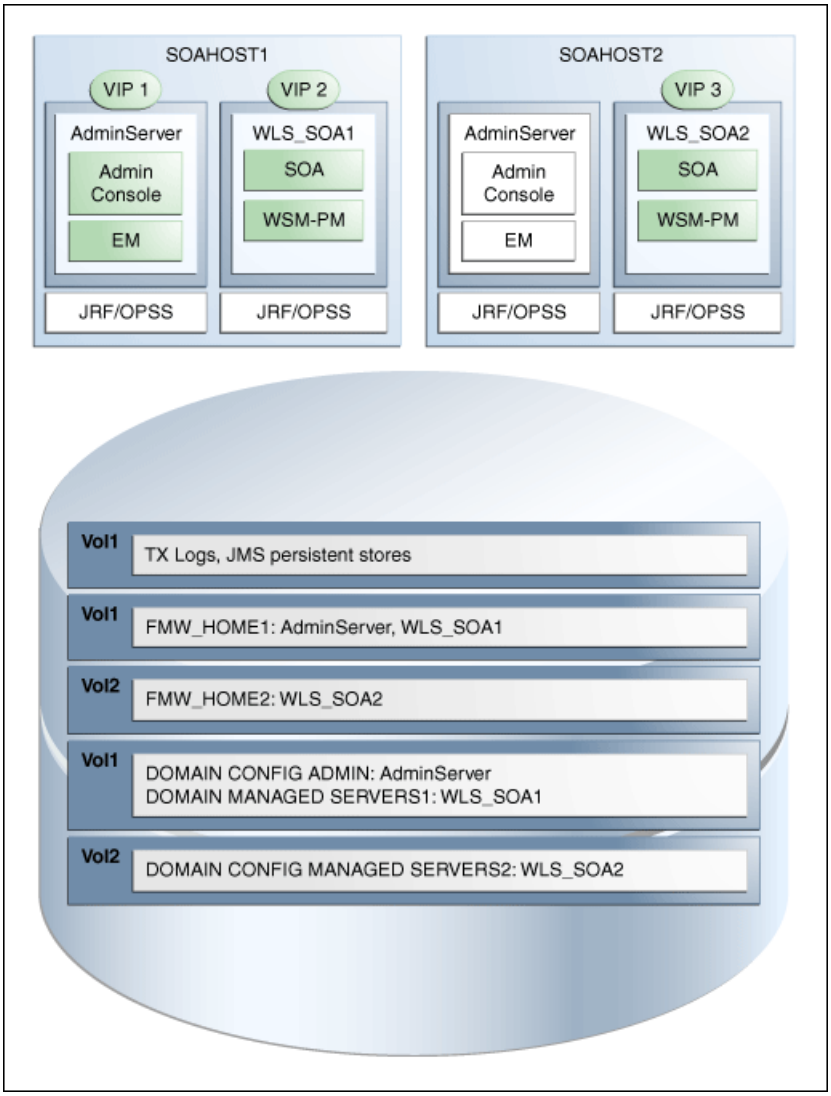


Table 4–2 summarizes the directory structure for an Oracle WebLogic Server domain that includes Oracle SOA Suite and Oracle WebCenter Content Managed Servers that share storage.

Table 4–2 Contents of Shared Storage

Server	Type of Data	Volume in Shared Storage	Directory	Files
WLS_WCC1 and WLS_WCC2	Tx Logs	VOL1	*ORACLE_BASE*/admin/*domain_name*/*img_cluster_name*/tlogs/	The transaction directory is common (decided by WebLogic Server), but the files are separate.
WLS_WCC1 and WLS_WCC2	JMS Stores	VOL1	*ORACLE_BASE*/admin/*domain_name*/*img_cluster_name*/jms/	The transaction directory is common (decided by WebLogic Server), but the files are separate.
WLS_SOA1 and WLS_WCC1	WLS Install	VOL1	*MW_HOME*/	Individual in each volume, but both servers see the same directory structure (SOAHOST1 and WCCHOST1 mount VOL1).
WLS_SOA2 and WLS_WCC2	WLS Install	VOL2	*MW_HOME*/	Individual in each volume, but both servers see the same directory structure (SOAHOST2 and WCCHOST2 mount VOL2).
WLS_WCC1	WCC Install	VOL1	*MW_HOME*/wcc/	Individual in each volume, but both servers see the same directory structure.
WLS_WCC2	WCC Install	VOL2	*MW_HOME*/wcc/	Individual in each volume, but both servers see the same directory structure.
WLS_SOA1 or WLS_SOA2	Domain Config	VOL1	*ORACLE_BASE*/admin/*domain_name*/aserver/*domain_name*/	Used by only one server where the Administration Server is running.
WLS_WCC1	Domain Config	VOL1	*ORACLE_BASE*/admin/*domain_name*/mserver/*domain_name*/	Individual in each volume, but both servers see the same directory structure.
WLS_WCC2	Domain Config	VOL2	*ORACLE_BASE*/admin/*domain_name*/mserver/*domain_name*/	Individual in each volume, but both servers see the same directory structure.
WLS_WCC1 and WLS_WCC2	Web and Vault Files	VOL3	*ORACLE_BASE*/admin/*domain_name*/*wcc_cluster_name*/vault/	Directory for vault files on a separate volume.
WLS_WCC1 and WLS_WCC2	Web and Vault Files	VOL3	*ORACLE_BASE*/admin/*domain_name*/*wcc_cluster_name*/weblayout/	Directory for weblayout files on a separate volume.
WLS_IBR1	Inbound Refinery Files	VOL3	*ORACLE_BASE*/admin/*domain_name*/*ibr_cluster_name*/ibrn/	Directory for all Inbound Refinery files.

4.4 Configuring Shared Storage

Use the following commands to create and mount shared storage locations so that WCCHOST1 and WCCHOST2 can see the same location for binary installation in two separate volumes.

> **Note:** The user ID used to create a shared storage file system owns those files and has read, write, and execute privileges for them. Other users in the operating system group can read and execute the files, but they do not have write privileges. For more information about installation and configuration privileges, see the "Understanding Installation and Configuration Privileges and Users" section in the *Oracle Fusion Middleware Installation Planning Guide*.

In the commands, `nasfiler` is the shared storage filer.

From SOAHOST1 and from WCCHOST1:

```
mount nasfiler:/vol/vol1/ORACLE_BASE/product/fmw ORACLE_BASE/product/fmw -t nfs
```

From SOAHOST2 and from WCCHOST2:

```
mount nasfiler:/vol/vol2/ORACLE_BASE/product/fmw ORACLE_BASE/product/fmw -t nfs
```

If only one volume is available, you can provide redundancy for the binaries by using two different directories in the shared storage and mounting them to the same directory in the Oracle WebCenter Content servers.

From SOAHOST1 and from WCCHOST1:

```
mount nasfiler:/vol/vol1/ORACLE_BASE/product/fmw1 ORACLE_BASE/product/fmw -t nfs
```

From SOAHOST2 and from WCCHOST2:

```
mount nasfiler:/vol/vol1/ORACLE_BASE/product/fmw2 ORACLE_BASE/product/fmw -t nfs
```

The following commands show how to share the Oracle WebCenter Content TX logs location across different nodes.

From WCCHOST1:

```
mount nasfiler:/vol/vol1/ORACLE_BASE/admin/domain_name/wcc_cluster_name/tlogs
ORACLE_BASE/admin/domain_name/wcc_cluster_name/tlogs -t nfs
```

From WCCHOST2:

```
mount nasfiler:/vol/vol1/ORACLE_BASE/admin/domain_name/wcc_cluster_name/tlogs
ORACLE_BASE/admin/domain_name/wcc_cluster_name/tlogs -t nfs
```

The following commands show how to share the Imaging `jms` location across different nodes.

From WCCHOST1:

```
mount nasfiler:/vol/vol1/ORACLE_BASE/admin/domain_name/img_cluster_name/jms
ORACLE_BASE/admin/domain_name/img_cluster_name/jms  -t nfs
```

From WCCHOST2:

```
mount nasfiler:/vol/vol1/ORACLE_BASE/admin/domain_name/img_cluster_name/jms
ORACLE_BASE/admin/domain_name/img_cluster_name/jms  -t nfs
```

The following commands show how to share WebCenter Content and Inbound Refinery files across different nodes.

```
mount nasfiler:/vol/vol3/ORACLE_BASE/admin/wcdomain/wcc_cluster_name/vault
-t nfs -o rw,bg,hard,vers=3

mount nasfiler:/vol/vol3/ORACLE_BASE/admin/wcdomain/ibr_cluster_name/
nfs -o rw,bg,hard,vers=3
```

Validating the Shared Storage Configuration

Ensure that you can read and write files to the newly mounted directories by creating a test file in the shared storage location you just configured.

For example:

```
cd newly_mounted_directory
touch testfile
```

Verify that the owner and permissions are correct:

```
ls -l testfile
```

Then remove the file:

```
rm testfile
```

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

```
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.

5

Preparing the Database for an Enterprise Deployment

This chapter describes procedures for preparing your database for an Oracle WebCenter Content enterprise deployment. The procedures include initial setup of the database, loading the metadata repository, and backing up the database.

This chapter includes the following topics:

5.1 Overview of Preparing the Database for an Enterprise Deployment

For the Oracle WebCenter Content 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 Oracle SOA Suite and Oracle WebCenter Content components. This database is separate from the Oracle Identity Management database, which is used in the Oracle Identity Management enterprise deployment 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 distribution or from the location listed in Table 2–2. For the enterprise topology, an Oracle Real Application Clusters (RAC) database is highly recommended.

When you configure the Oracle SOA Suite components, the Fusion Middleware Configuration Wizard will prompt you to enter the information for connecting to the database that contains the metadata repository.

5.2 Database Requirements

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

- Section 5.2.1, "Database Host Requirements"
- Section 5.2.2, "Supported Database Versions"
- Section 5.2.3, "Initialization Parameters"

5.2.1 Database Host Requirements

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

- **Oracle Clusterware**

  For 11*g* Release 1 (11.1) for Linux, refer to the *Oracle Clusterware Installation Guide for Linux*.

- **Oracle Real Application Clusters**

  For 11*g* Release 1 (11.1) for Linux, refer to the *Oracle Real Application Clusters Installation Guide for Linux and UNIX*. For 10*g* Release 2 (10.2) for Linux, refer to the *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Installation Guide for Linux*.

- **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.

5.2.2 Supported Database Versions

Oracle WebCenter Content requires the presence of a supported database and schemas. To check if your database is certified or to see all certified databases, refer to the "Oracle Fusion Middleware 11*g* Release 1 (11.1.1.*x*)" product area on the Oracle Fusion Middleware Supported System Configurations page:

`http://www.oracle.com/technetwork/middleware/ias/downloads/fusion-certification-100350.html`

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

```
SELECT VERSION FROM SYS.PRODUCT_COMPONENT_VERSION WHERE PRODUCT LIKE "Oracle%";
```

Notes:

- Oracle WebCenter Content requires that the database used to store its metadata (either 10*g* or 11*g*) supports the AL32UTF8 character set. Check the database documentation for information on choosing a character set for the database.
- For Oracle WebCenter Content enterprise deployments, Oracle recommends using GridLink data sources to connect to Oracle RAC databases. To use the Oracle Single Client Access Name (SCAN) feature with GridLink, the Oracle RAC database version must be Oracle Database 11*g*R2 (11.2 or later, Enterprise Edition).

5.2.3 Initialization Parameters

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

Table 5–1 Required Initialization Parameters

Configuration	Parameter	Required Value	Parameter Class
Oracle SOA Suite	`PROCESSES`	400 or greater	Static
Oracle WebCenter Content	`PROCESSES`	100 or greater	Static
Oracle SOA Suite and Oracle WebCenter Content	`PROCESSES`	500 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:

```
SHOW PARAMETER processes
```

Set the initialization parameter using the following command:

```
ALTER SYSTEM SET processes=500 open_cursors=500 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. For details on parameter files, server parameter files, and how to change parameter value, see the *Oracle Database Administrator's Guide*.

5.3 Creating Database Services

When multiple Oracle Fusion Middleware products are sharing the same database, each product should be configured to connect to a separate, dedicated database service. For more information about connecting to Oracle databases using services, see "Overview of Connecting to Oracle Database Using Services and VIP Addresses" in the *Oracle Real Application Clusters Administration and Deployment Guide*. For complete instructions on creating and managing database services, see "Introduction to Automatic Workload Management" in the *Oracle Real Application Clusters Administration and Deployment Guide*.

Run-time connection load-balancing requires configuring Oracle RAC Load Balancing Advisory with service-level goals for each service for which load balancing is enabled. You can configure the Oracle RAC Load Balancing Advisory for `SERVICE_TIME` or `THROUGHPUT`. Set the connection load-balancing goal to `SHORT`. For Oracle Database 10*g* or 11*g* Release 1 (11.1), use the `DBMS_SERVICE` package for this modification. For Oracle Database 11*g* Release 2 (11.2), use the Server Control Utility (SRVCTL) instead.

The Oracle WebCenter Content installation is configured to use the service `wccedg.mycompany.com`. For Oracle SOA Suite, using a service named `soaedg.mycompany.com` is recommended.

> **Note:** For simplicity, the data source configuration screens in this guide use the same service name (`wccedg.mycompany.com`).

This section includes the following topics:

- Section 5.3.1, "Creating Database Services for 10g and 11g Release 1 (11.1) Databases"
- Section 5.3.2, "Creating Database Services for 11g Release 2 (11.2) Databases"

5.3.1 Creating Database Services for 10g and 11g Release 1 (11.1) Databases

You can create and modify 10*g* and 11*g* Release 1 (11.1) database services using the `DBMS_SERVICE` package.

To create and modify a 10g or 11.1 database service:

1. Log in to SQL*Plus, and create the service:

```
sqlplus "sys/password as sysdba"

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

Notes:

- For the service name of an Oracle RAC database, use lowercase letters, followed by the domain name. For example:

  `wccedg.mycompany.com`

- Enter the `EXECUTE DBMS_SERVICE` command shown on a single line.

  For more information about the `DBMS_SERVICE` package, see the *Oracle Database PL/SQL Packages and Types Reference*.

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

```
srvctl add service -d ecmdb -s wccedg.mycompany.com -r ecmdb1,ecmdb2
```

3. Start the service:

```
srvctl start service -d ecmdb -s wccedg.mycompany.com
```

Note: For complete instructions on creating and managing database services with SRVCTL, see "Administering Services with SRVCTL" in the *Oracle Real Application Clusters Administration and Deployment Guide*.

4. Modify the service for the appropriate service goals, with either of these `EXECUTE` commands:

```
SQL>EXECUTE DBMS_SERVICE.MODIFY_SERVICE (service_name =>
'wccedg.mycompany.com',goal => DBMS_SERVICE.GOAL_THROUGHPUT, clb_goal =>DBMS_
SERVICE.CLB_GOAL_SHORT);

SQL>EXECUTE DBMS_SERVICE.MODIFY_SERVICE (service_name =>
'wccedg.mycompany.com', goal => DBMS_SERVICE.GOAL_SERVICE_TIME, clb_goal
=>DBMS_SERVICE.CLB_GOAL_SHORT);
```

5.3.2 Creating Database Services for 11g Release 2 (11.2) Databases

You can create and modify 11*g* Release 2 (11.2) database services using the `srvctl` command of SRVCTL.

To create and modify an 11.2 database service:

1. Log in to SQL*Plus and create the service:

```
sqlplus "sys/password as sysdba"

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

Notes:

- For the service name of an Oracle RAC database, use lowercase letters, followed by the domain name. For example:

  `wccedg.mycompany.com`

- Enter the `EXECUTE DBMS_SERVICE` command shown on a single line.

  For more information about the `DBMS_SERVICE` package, see the *Oracle Database PL/SQL Packages and Types Reference*.

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

```
srvctl add service -d ecmdb -s wccedg.mycompany.com -r ecmdb1,ecmdb2
```

3. Start the service:

```
srvctl start service -d ecmdb -s wccedg.mycompany.com
```

Note: For complete instructions on creating and managing database services with SRVCTL, see "Administering Services with SRVCTL" in the *Oracle Real Application Clusters Administration and Deployment Guide.*

4. Modify the service for the appropriate service goal, with either of these `srvctl` commands:

```
srvctl modify service -d ecmdb -s wccedg.mycompany.com -B SERVICE_TIME -j SHORT

srvctl modify service -d ecmdb -s wccedg.mycompany.com -B  THROUGHPUT  -j SHORT
```

For more information about the different service definitions, see "Load Balancing Advisory" in the *Oracle Real Application Clusters Administration and Deployment Guide*.

5.4 Loading the Oracle Fusion Middleware Metadata Repository in the Oracle RAC Database

The Repository Creation Utility (RCU) is available from the RCU distribution. The RCU used to seed the database must match the patch set level of the Oracle WebCenter Content installation. This means that if you install Oracle WebCenter Content 11*g* Release 1 (11.1.1.7) in this enterprise deployment, you must use RCU 11*g* Release 1 (11.1.1.7).

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

1. Open the Repository Creation Utility (RCU) distribution, and then start RCU from the bin directory in the RCU home directory:

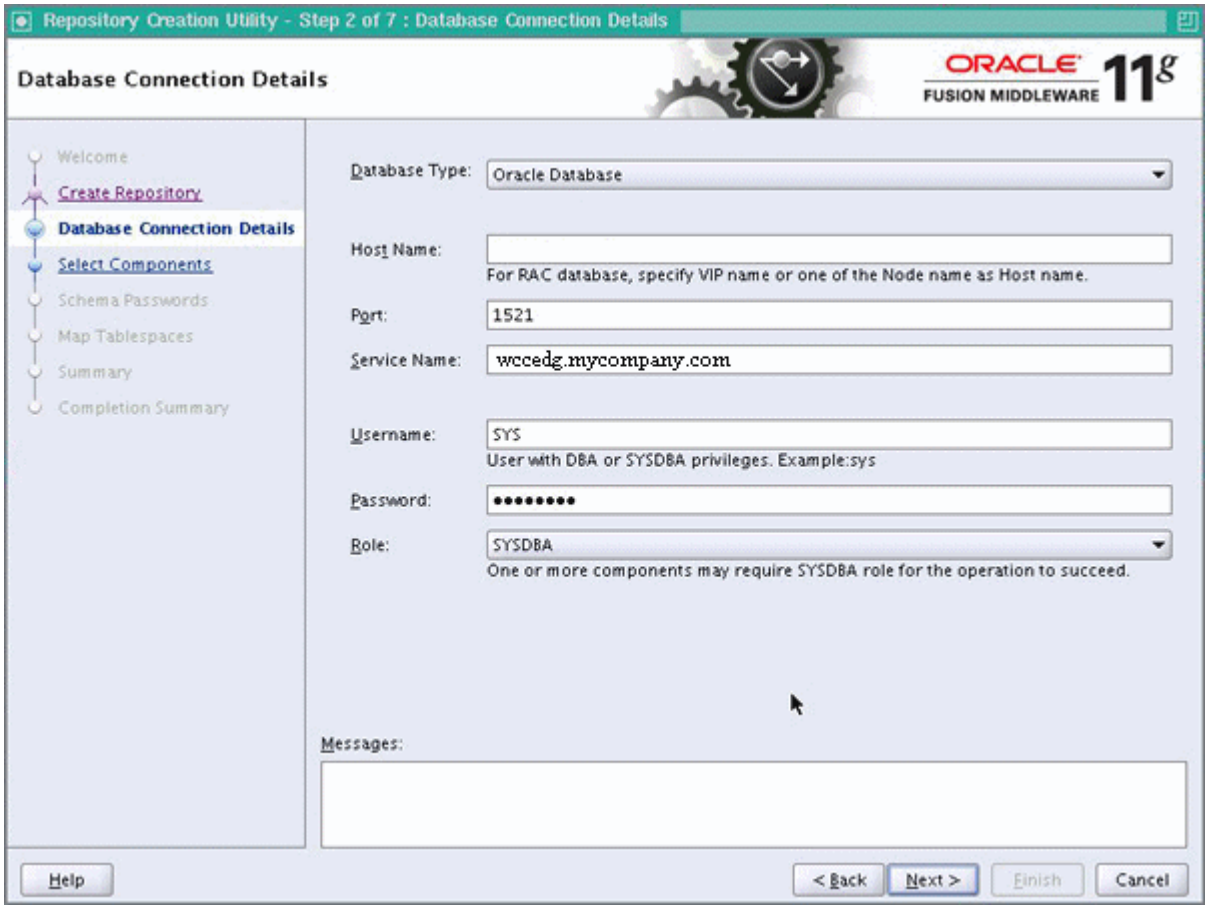
```
cd RCU_HOME/bin
./rcu
```

2. In the Welcome screen (if displayed), click **Next**.

3. In the Create Repository screen, select **Create** to load component schemas into a database. Click **Next**.
4. In the Database Connection Details screen (Figure 5–1), enter the connect information for your database:
   - **Database Type**: Select **Oracle Database**.
   - **Host Name**: Specify the name of the node on which the database resides. For the Oracle RAC database, specify the VIP name or one of the node names as the host name; for example, CUSTDBVIP1.
   - **Port**: Specify the listen port number for the database: 1521.
   - **Service Name**: Specify the service name of the database (`wccedg.mycompany.com`).
   - **Username**: Specify the name of the user with `DBA` or `SYSDBA` privileges: `SYS`.
   - **Password**: Enter the password for the `SYS` user.
   - **Role**: Choose the database user's role from the list: **SYSDBA** (required by the `SYS` user).

   Click **Next**.

Figure 5–1 Database Connection Details Screen



5. If your database is not using the UTF-8 character set, you will see a warning message indicating that there may be data loss if you are going to use the database for multilingual support. If you are not planning to use multilingual support, then you can click **Ignore**. Otherwise, click **Stop**.
6. In the Select Components screen (Figure 5–2), do the following:
   - Select **Create a new Prefix**, and enter a prefix to use for the database schemas, for example `DEV` or `PROD`. You can specify up to six characters as a prefix. Prefixes are used to create logical groupings of multiple repositories in a database. For more information, see the *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 components:
     – AS Common Schemas:

       - **Metadata Services**

     – WebCenter Content:

       - **Oracle WebCenter Content Server - Complete**

       - **Oracle WebCenter Content: Imaging**

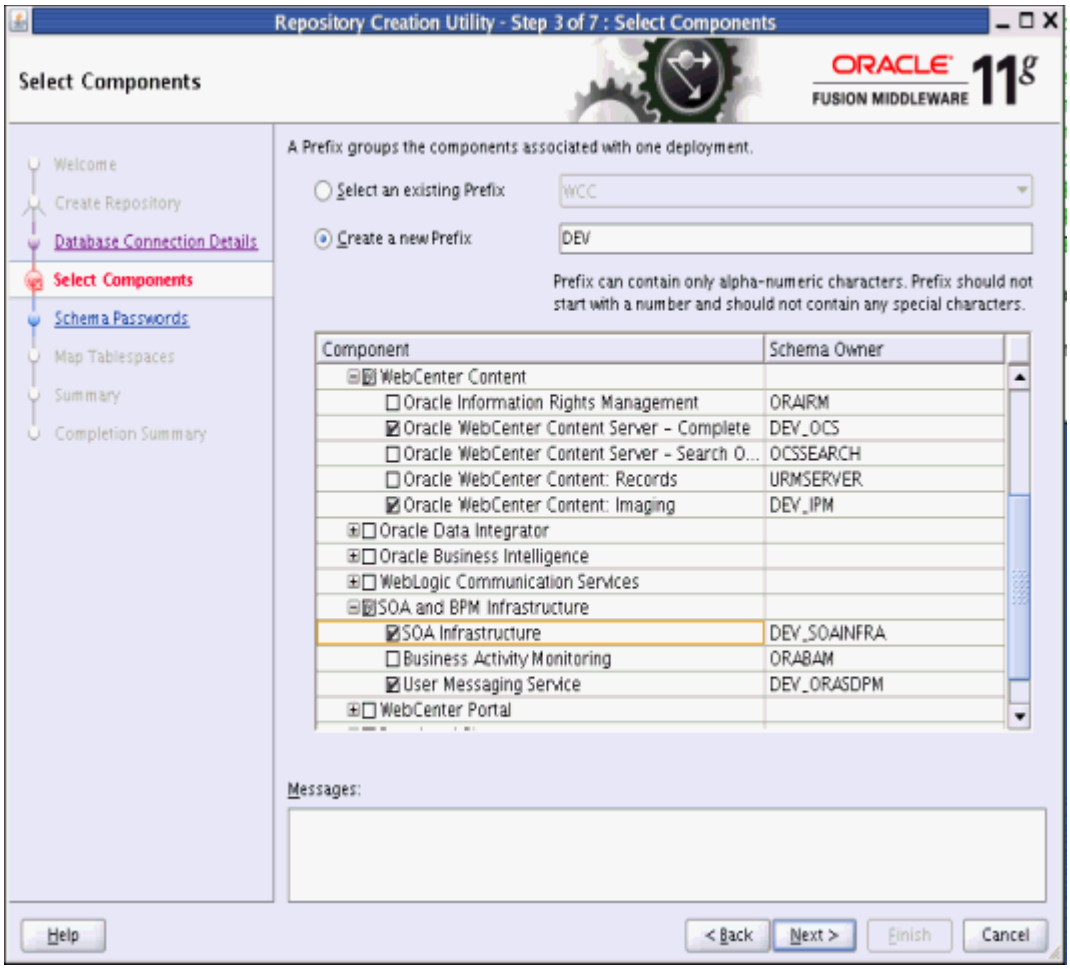
     – SOA and BPM Infrastructure:

       - **SOA Infrastructure**

       - **User Messaging** (automatically selected with SOA Infrastructure)

   Click **Next**.

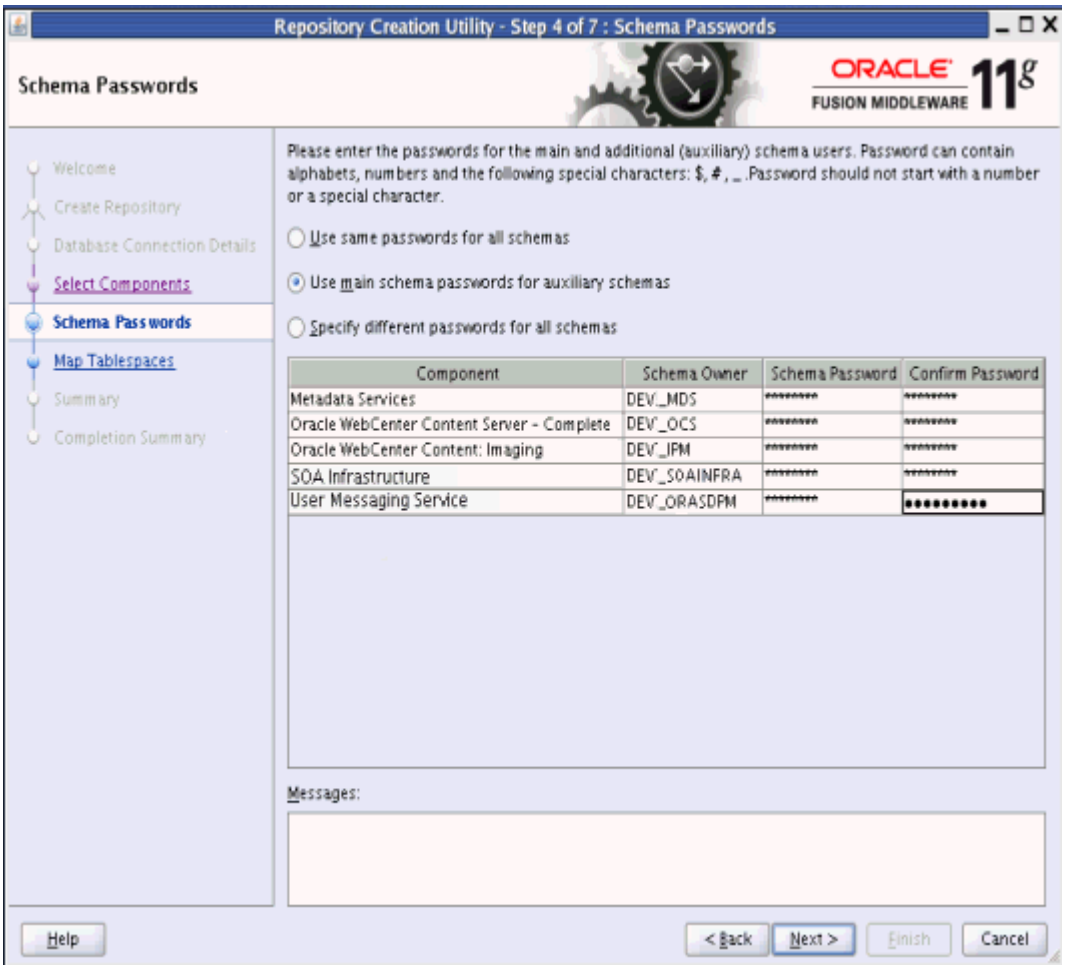
Figure 5–2 Select Components Screen



7. In the Schema Passwords screen (Figure 5–3), select **Use main schema passwords for auxiliary schemas**, and click **Next**. In the subsequent screen refresh, enter the schema passwords for all components.

   Tip: Note the name of the schema because the upcoming steps require this information.

Figure 5–3 Schema Passwords Screen



8. In the Map Tablespaces screen, choose the tablespaces for the selected components, and click **Next**.

   A confirmation dialog opens stating that any tablespace that does not already exist in the selected schema will be created. Click **OK** to acknowledge this message.

9. In the Summary screen, click **Create**.

10. In the Completion Summary screen, click **Close**.

11. Verify that the required schemas were created successfully by connecting to the database with the new user added:

    ```
    ORACLE_HOME/bin/sqlplus
    ```

    Log in as the `DEV_OCS` user, and enter the password. You can perform a simple verification by querying the schema version registry:

    ```
    -bash-3.00$ $ORACLE_HOME/bin/sqlplus DEV_OCS/password as SYSDBA
    SQL> SELECT version, status FROM schema_version_registry where owner =
    "DEV_OCS";

    VERSION STATUS
    ----------------------------------------
    11.1.1.7.0 VALID
    ```

Note: Oracle recommends using the database used for Oracle Identity Management (see Chapter 15, "Integrating with Oracle Identity Management") to store the Oracle WSM policies. It is therefore expected that you will use the Oracle Identity Management database information for the OWSM MDS schemas, which will be different from the database information used for the rest of the Oracle SOA Suite schemas. To create the required schemas in the database, repeat the steps above using the Oracle Identity Management database information, but select only **AS Common Schemas: Metadata Services** in the Select Components screen (step 6).

5.5 Backing Up the Database

After you have loaded the metadata repository into your database, you should make a backup before installing the software for your enterprise deployment.

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 make a backup using operating system tools or RMAN for this purpose. It is recommended that you 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.

6

Installing the Software for an Enterprise Deployment

This chapter describes the software installations required for the Oracle WebCenter Content enterprise deployment reference topology. You install Oracle HTTP Server and then Oracle Fusion Middleware.

> **Note:** Before starting the setup process, read the *Oracle Fusion Middleware Release Notes* for your platform for additional installation and deployment information.

This chapter contains the following sections:

6.1 Overview of the Software Installation Process

The enterprise deployment software installation is divided into two parts. The first part covers the required web tier installations, while the second part addresses the required Oracle Fusion Middleware components. Later chapters describe the required configuration steps to create the Oracle WebCenter Content reference topology.

Obtaining the Software

For information about where to obtain the software, see "Obtain the Oracle Fusion Middleware Software" in the *Oracle Fusion Middleware Installation Planning Guide*.

Select one of the download locations and download Oracle WebCenter Content and Oracle SOA Suite. The ZIP archive files are saved to your system.

After you download each archive file, extract it into a directory of your choice on the machine where you are performing the installation.

Software to Install

Table 6–1 shows what software should be installed on each host or be accessible from each host.

Table 6–1 Software To Be Installed On Each Host or Accessible From Each Host

Hosts	Oracle HTTP Server	Oracle WebLogic Server	Oracle SOA Suite	Oracle WebCenter Content
WEBHOST1	X			
WEBHOST2	X			
SOAHOST1		X	X	X
SOAHOST2		X	X	X
WCCHOST1		X	X	X
WCCHOST2		X	X	X

6.2 Installing Oracle HTTP Server

This section covers these topics:

- Section 6.2.1, "Prerequisites to Installing Oracle HTTP Server"
- Section 6.2.2, "Installing Oracle HTTP Server on WEBHOST1 and WEBHOST2"
- Section 6.2.3, "Backing Up the Installation"

6.2.1 Prerequisites to Installing Oracle HTTP Server

Prior to installing Oracle HTTP Server (OHS), check that your machines meet the following requirements:

- Ensure that the system, patch, kernel, and other requirements are met as specified in the *Oracle Fusion Middleware Installation Guide for Oracle Web Tier*.
- Because Oracle HTTP Server is installed on port 7777 by default, you must make sure 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:

  ```
  netstat -an | grep 7777
  ```

  You must free port 7777 if it is in use.
- 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.
- Before starting the installation, make sure that the following environment variables are not set:
  - LD_ASSUME_KERNEL
  - ORACLE_INSTANCE

6.2.2 Installing Oracle HTTP Server on WEBHOST1 and WEBHOST2

As described in Chapter 4, "Preparing the File System for an Enterprise Deployment," you install Oracle Fusion Middleware in at least two storage locations for redundancy.

1. Start the installer for Oracle HTTP Server from the installation media:

   ```
   ./runInstaller
   ```

2. 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 the screen to execute `createCentralInventory.sh` as `root`.

   Click **OK**.

3. In the Welcome screen, click **Next**.

4. In the Install Software Updates screen, choose **Skip Software Updates** and click **Next**.

5. In the Select Installation Type screen, select **Install - Do Not Configure**, and click **Next**.

6. In the Prerequisite Checks screen, verify that all checks complete successfully, and click **Next**.

7. In the Specify Installation Location screen, specify the following values:

   - **Fusion Middleware Home Location** (installation location): *ORACLE_BASE*`/product/fmw`
   - **Oracle Home Location Directory:** `web`

   Click **Next**.

8. In the Specify Security Updates screen, choose whether you want to receive security updates from Oracle support and if you do, enter your email address.

9. In the Installation Summary screen, review the selections to ensure they are correct. If they are not, click **Back** to modify selections on previous screens. When you are ready, click **Install**.

   On UNIX systems, if prompted to run the `oracleRoot.sh` script, make sure you run it as the `root` user.

   The Oracle HTTP Server software is installed.

10. In the Installation Completed screen, click **Finish** to exit.

11. Validate the installation by verifying that the following directories appear in the *MW_HOME* directory after you install Oracle HTTP Server:

    - `oracle_common`
    - `web`

6.2.3 Backing Up the Installation

The Middleware home should be backed up now from WEBHOST1 (make sure no server is running at this point):

```
tar -cvpf fmwhomeback.tar ORACLE_BASE/product/fmw HOME/oraInventory
```

6.3 Installing Oracle Fusion Middleware

This section describes how to install the required Oracle Fusion Middleware software for the Oracle WebCenter Content enterprise deployment reference topology. The software components to be installed consist of the WebLogic Server home (WL_HOME) and Oracle home (ORACLE_HOME). As described in Chapter 4, "Preparing the File System for an Enterprise Deployment," you install Oracle Fusion Middleware in at least two storage locations for redundancy.

> **Note:** Before starting the setup process, read the *Oracle Fusion Middleware Release Notes* for your platform for additional installation and deployment information.

This section covers these topics:

- Section 6.3.1, "Installing Oracle WebLogic Server and Creating the Middleware Home"
- Section 6.3.2, "Installing Oracle Fusion Middleware Components"
- Section 6.3.3, "Backing Up the Installation"

6.3.1 Installing Oracle WebLogic Server and Creating the Middleware Home

To install Oracle WebLogic Server on SOAHOST1 and SOAHOST2:

> **Note:** For information about running the generic installer for installing WebLogic Server on 64-bit platforms using a 64-bit JDK, see the section "Installing WebLogic Server on 64-Bit Platforms Using a 64-Bit JDK" in the *Oracle Fusion Middleware Installation Guide for Oracle WebLogic Server*.

1. Start the installer for Oracle WebLogic Server:

   ```
   $ ./wls1036_linux32.bin
   ```

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 *ORACLE_BASE*`/product/fmw`.

     ORACLE_BASE is the base directory under which Oracle products are installed. The recommended value is `/u01/app/oracle`. For more information, see Section 4.3, "About Recommended Locations for the Different Directories."

   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_*version* SDK**, and click **Next**.
8. In the Choose Product Installation Directories screen, accept the directories ***ORACLE_BASE*/product/fmw/wlserver_10.3** and ***ORACLE_BASE*/product/fmw/coherence_3.7**, and click **Next**.
9. In the Installation Summary screen, click **Next**.

   The Oracle WebLogic Server software is installed.
10. In the Installation Complete screen, clear the **Run Quickstart** checkbox, and click **Done**.
11. Validate the installation by verifying that the following directories and files appear in the *MW_HOME*/ directory after installing Oracle WebLogic Server:
    - coherence_*version*
    - jrockit-jdk1.6.0_*version*
    - modules
    - registry.xml
    - utils
    - domain-registry.xml
    - logs
    - ocm.rsp
    - registry.dat
    - wlserver_10.3

6.3.2 Installing Oracle Fusion Middleware Components

This section covers these topics:

- Section 6.3.2.1, "Installing Oracle SOA Suite"
- Section 6.3.2.2, "Installing Oracle WebCenter Content"

6.3.2.1 Installing Oracle SOA Suite

To install Oracle SOA Suite on SOAHOST1 and SOAHOST2:

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 SOA Suite from the installation media:

   ```
   ./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, *ORACLE_BASE*/product/fmw/jrockit-jdk1.6.0_*version*. For more information, see Section 6.3.1, "Installing Oracle WebLogic Server and Creating the Middleware Home."

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 **OK**.

   Follow the instructions on screen to execute `createCentralInventory.sh` as `root`.

   Click **OK**.

4. In the Welcome screen, click **Next**.

5. In the Install Software Updates screen, select **Skip Software Updates** and click **Next**.

6. In the Prerequisite Checks screen, verify that all checks complete successfully, and click **Next**.

7. In the Specify Installation Location screen (Figure 6–1), provide the installation location for Oracle SOA Suite. Select the previously installed Middleware home from the drop-down list. For the Oracle home directory, enter the directory name (`soa`).

   Note: Since the installation is performed on a shared storage, the Middleware home is accessible and used by the Oracle WebCenter Content servers in WCCHOST1 and WCCHOST2.

Figure 6–1 Specify Installation Location Screen in Installer Wizard

   Click **Next** when you are done.

8. In the Application Server screen, make sure **WebLogic Server** is selected (which is the default), and click **Next**.

9. In the Installation Summary screen, click **Install**.

   The Oracle SOA Suite software is installed.

10. In the Installation Complete screen, click **Finish**.

11. Validate the installation by verifying that the following directories and files appear in the *MW_HOME/* directory after installing Oracle WebLogic Server, Oracle Fusion Middleware, and Oracle SOA Suite:

    - coherence_*version*
    - jrockit-jdk*version*
    - modules
    - oracle_common
    - registry.xml
    - utils
    - domain-registry.xml
    - logs
    - ocm.rsp
    - registry.dat
    - soa
    - wlserver_10.3

6.3.2.2 Installing Oracle WebCenter Content

When you install the Oracle WebCenter Content software, you install product bits for the following feature sets on your system:

- Oracle WebCenter Content
- Oracle WebCenter Content: Imaging
- Oracle WebCenter Content: Records
- Oracle WebCenter Content: Inbound Refinery

To install Oracle WebCenter Content on SOAHOST1 and SOAHOST2:

> **Note:** Since the installation is performed on a shared storage, the Middleware home is accessible and used by the Oracle WebCenter Content servers in WCCHOST1 and WCCHOST2.

1. Start the installer for Oracle WebCenter Content:

   ```
   ./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, *ORACLE_BASE*/product/fmw/jrockit-jdk1.6.0_*version*. For more information, see Section 6.3.1, "Installing Oracle WebLogic Server and Creating the Middleware Home."

2. In the Welcome screen, click **Next**.

3. In the Install Software Updates screen, select **Skip Software Updates**, and click **Next**.
4. In the Prerequisite Checks screen, verify that all checks complete successfully, and click **Next**.
5. In the Specify Installation Location screen, provide the installation location for Oracle WebCenter Content. Select the previously installed Middleware home from the drop-down list. For the Oracle home directory, enter the directory name (`wcc`).

   Click **Next** when you are done.
6. In the Application Server screen, make sure **WebLogic Server** is selected (which is the default), and click **Next**.
7. In the Installation Summary screen, click **Install**.

   The Oracle WebCenter Content software is installed.
8. In the Installation Complete screen, click **Finish**.
9. Validate the installation by verifying that the following directories and files appear in the *MW_HOME/* directory after installing Oracle WebLogic Server, Oracle Fusion Middleware, Oracle SOA Suite, and Oracle WebCenter Content:
   - coherence_*version*
   - jrockit-jdk*version*
   - modules
   - oracle_common
   - registry.xml
   - utils
   - domain-registry.xml
   - logs
   - ocm.rsp
   - registry.dat
   - soa
   - wlserver_10.3
   - wcc

6.3.3 Backing Up the Installation

The Middleware home should be backed up now from SOAHOST1 (make sure that you stop the servers first):

```
tar -cvpf fmwhomeback.tar ORACLE_BASE/product/fmw
```

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

7 Configuring Oracle Web Tier

This chapter describes how to configure Oracle Web Tier to support the Oracle Fusion Middleware Oracle WebCenter Content implementation.

This chapter contains the following sections:

7.1 Overview of Configuring Oracle Web Tier

Before configuring the Oracle Web Tier software, you need to install it on WEBHOST1 and WEBHOST2, as described in Section 6.2, "Installing Oracle HTTP Server," and define the instance home, instance name, and Oracle HTTP Server component name, as described in Section 7.2, "Running the Configuration Wizard to Configure Oracle HTTP Server."

Then you can validate the installation and configure the load balancer to route all HTTP requests to WEBHOST1 and WEBHOST2.

Finally, you can configure the virtual hosts by defining directives in configuration files. You created the following virtual host names when you configured the load balancer in Section 3.3, "Configuring the Load Balancers":

- wcc.mycompany.com
- admin.mycompany.com
- soainternal.mycompany.com

7.2 Running the Configuration Wizard to Configure Oracle HTTP Server

The steps for configuring the Oracle Web Tier are the same for both WEBHOST1 and WEBHOST2.

To configure Oracle Web Tier:

1. Change the directory to the location of the Oracle Fusion Middleware Configuration Wizard:

   ```
   cd ORACLE_BASE/product/fmw/web/bin
   ```

2. Start the Configuration Wizard:

   ```
   ./config.sh
   ```

3. In the Welcome screen, click **Next**.

4. In the Configure Components screen, select **Oracle HTTP Server** and unselect **Associate Selected Components with WebLogic Domain**. Make sure that Oracle Web Cache is *not* selected.

   Click **Next**.

5. In the Specify Component Details screen, specify the following values:

   - **Instance Home Location**: *ORACLE_BASE*`/admin/webn`
   - **AS Instance Name**: `webn`
   - **OHS Component Name**: `ohsn`

   (where *n* is a sequential number for your installation; for example, `1` for WEBHOST1, `2` for WEBHOST2, and so on)

   > **Note:** The Oracle HTTP Server instance names on WEBHOST1 and WEBHOST2 must be different.

   Click **Next**.

6. In the Configure Ports screen, select **Specify Ports using Configuration file**, then select a file name, and then click **View/Edit**.

   In high-availability implementations, it is not mandatory for all of the ports used by the various components to be synchronized across hosts; however, it makes the enterprise deployment much simpler. You can bypass automatic port configuration by specifying the ports in a file.

   The file will look like this:

   ```
   [OHS]
   #Listen port for OHS component
   OHS Port = 7777

   [OPMN]
   #Process Manager Local port no
   OPMN Local Port = 1880
   ```

   You can find a sample `staticports.ini` file on installation disk 1 in the `stage/Response/` directory.

   Click **Next**.

7. In the Specify Security Updates screen, choose whether you want to receive security updates from Oracle support and if you do, enter your email address.

8. In the Installation Summary screen, review the selections to ensure they are correct. If they are not, click **Back** to modify selections on previous screens. When you are ready, click **Configure**.

9. Multiple configuration assistants are launched in succession; this process can be lengthy. When it completes, click **Next**, and the Installation Complete screen appears.

10. In the Installation Completed screen, click **Finish** to exit.

7.3 Validating the Installation

Once the installation is completed, check that it is possible to access the Oracle HTTP Server home page using the following URL:

```
http://webhost1.mycompany.com:7777/
http://webhost2.mycompany.com:7777/
```

7.4 Configuring Oracle HTTP Server with the Load Balancer

Configure your load balancer to route all HTTP requests to the hosts running Oracle HTTP Server (WEBHOST1, WEBHOST2). You do not need to enable sticky sessions (insert cookie) on the load balancer when Oracle HTTP Server is front-ending Oracle WebLogic Server. You need sticky sessions if you are going directly from the load balancer to Oracle WebLogic Server, which is not the case in the topology described in this guide. Also, you should set monitors for HTTP.

The instructions for this configuration will vary depending on which load balancer you use. See your load balancer documentation for specific instructions.

7.5 Defining Virtual Hosts

The reference topology in this guide requires that you define a set of virtual hosts for the Oracle HTTP Server. For each virtual host, you will later define a set of specific URLs that will route requests to the proper Administration Server or Managed Server in the WebLogic Server domain.

This section contains the following topics:

- Section 7.5.1, "Creating *_vh.conf Files to Define <VirtualHost> Directives"
- Section 7.5.2, "Validating the Configuration"

7.5.1 Creating *_vh.conf Files to Define <VirtualHost> Directives

Define each virtual host in its own `*_vh.conf` file. This will make it easy to manage the URLs for each virtual host you define.

Create the following new files to define the `<VirtualHost>` directives:

- `admin_vh.conf`
- `soainternal_vh.conf`
- `wcc_vh.conf`

Create the new files in each of the following directories:

ORACLE_INSTANCE`/config/`*instance_name*`/config/OHS/ohs1/moduleconf/`

ORACLE_INSTANCE`/config/`*instance_name*`/config/OHS/ohs2/moduleconf/`

To define each virtual host in its own `*_vh.conf` file:

1. Create the `admin_vh.conf` file, and add the following directive:

```
NameVirtualHost *:7777
<VirtualHost *:7777>
    ServerName admin.mycompany.com:80
    ServerAdmin you@your.address
    RewriteEngine On
    RewriteOptions inherit
    RewriteRule ^/console/jsp/common/logout.jsp "/oamsso/logout.html?end_
url=/console" [R]
</VirtualHost>
```

   If the steps in Chapter 15, "Integrating with Oracle Identity Management," have not been completed, then comment out the `RewriteRule` lines until the integration has been completed.

2. Create the `soainternal_vh.conf` file, and add the following directive:

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

3. Create the `wcc_vh.conf` file, and add the following directive:

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

4. Restart both Oracle HTTP Servers:

```
cd ORACLE_BASE/admin/instance_name/bin
opmnctl stopall
opmnctl startall
```

> **Note:** Values such as `wcc.mycompany.com:443`, `ServerAdmin you@your.address`, and `admin.mycompany.com:80`, are only examples. Enter values based on your actual environment.

7.5.2 Validating the Configuration

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

- `https://wcc.mycompany.com/index.html`
- `http://admin.mycompany.com/index.html`
- `http://soainternal.mycompany.com/index.html`

If you cannot access these URLs, check to ensure that you completed the procedure in Section 3.3, "Configuring the Load Balancers," correctly.

8

Creating a Domain for an Enterprise Deployment

This chapter describes how to create a domain using the Oracle Fusion Middleware Configuration Wizard, Oracle WebLogic Server Administration Console, and Oracle Enterprise Manager Fusion Middleware Control. You can extend the domain to add Fusion Middleware (FMW) components: Oracle SOA Suite, Oracle WebCenter Content and, optionally, Oracle WebCenter Content: Imaging and Oracle WebCenter Content: Inbound Refinery. This will be addressed in later chapters in this document.

> **Note:** Before starting the setup process, read the *Oracle Fusion Middleware Release Notes* for your platform for additional installation and deployment information.

This chapter contains the following sections:

8.1 Overview of Creating a Domain

Table 8–1 lists the steps for creating an Oracle WebLogic Server domain, including post-configuration tasks.

Table 8–1 Steps for Creating a WebLogic Server domain

Step	Description	More Information
Enabling VIP1 in SOAHOST1	Enable ADMINVHN for the SOAHOST1 host	Section 8.2, "Enabling VIP1 on SOAHOST1"
Create a WebLogic Server Domain	Run the Configuration Wizard to create a WebLogic Server domain.	Section 8.3, "Running the Configuration Wizard on SOAHOST1 to Create a Domain"
Post-Configuration and Verification Tasks	Follow the instructions for post-configuration and validation tasks.	Section 8.4, "Post-Configuration and Verification Tasks"
Configure the Oracle HTTP Server with the WebLogic Server domain	Configure Oracle HTTP Server with the WebLogic Server domain, and validate the configuration.	Section 8.5, "Configuring Oracle HTTP Server for the WebLogic Server Domain"
Back Up the Domain	Back up the newly configured WebLogic Server domain.	Section 8.6, "Backing Up the WebLogic Server Domain Configuration"

After this domain is created and configured, you can extend the domain to include Oracle SOA Suite and Oracle WebCenter Content components, as described in the next chapters.

8.2 Enabling VIP1 on SOAHOST1

This step is required for failover of the Administration Server, regardless of whether or not other Fusion Middleware components are installed later.

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

To enable the virtual IP, see Section 3.5, "Enabling Virtual IP Addresses for an Enterprise Deployment."

Check that the virtual hosts are enabled as Table 8–2 shows.

Table 8–2 Virtual Hosts

VIP	Enabled on Host
ADMINVHN.mycompany.com	SOAHOST1
SOAHOST1VHN1.mycompany.com	SOAHOST1
SOAHOST2VHN1.mycompany.com	SOAHOST2
WCCHOST1VHN1.mycompany.com	WCCHOST1
WCCHOST2VHN1.mycompany.com	WCCHOST2

Note: This is the DNS name associated with the floating IP address. It is not the DNS name of the virtual host configured on the load balancer.

8.3 Running the Configuration Wizard on SOAHOST1 to Create a Domain

Run the Oracle Fusion Middleware Configuration Wizard from the Oracle Common home directory to create a domain containing the Administration Server. You will extend the domain to contain other components later.

1. Ensure that the database where you installed the repository is running. For Oracle RAC databases, it is recommended that all instances are running, so that the validation checks later on becomes more reliable.
2. On SOAHOST1, change the directory to the location of the Oracle Fusion Middleware Configuration Wizard (created in Chapter 6, "Installing the Software for an Enterprise Deployment"):

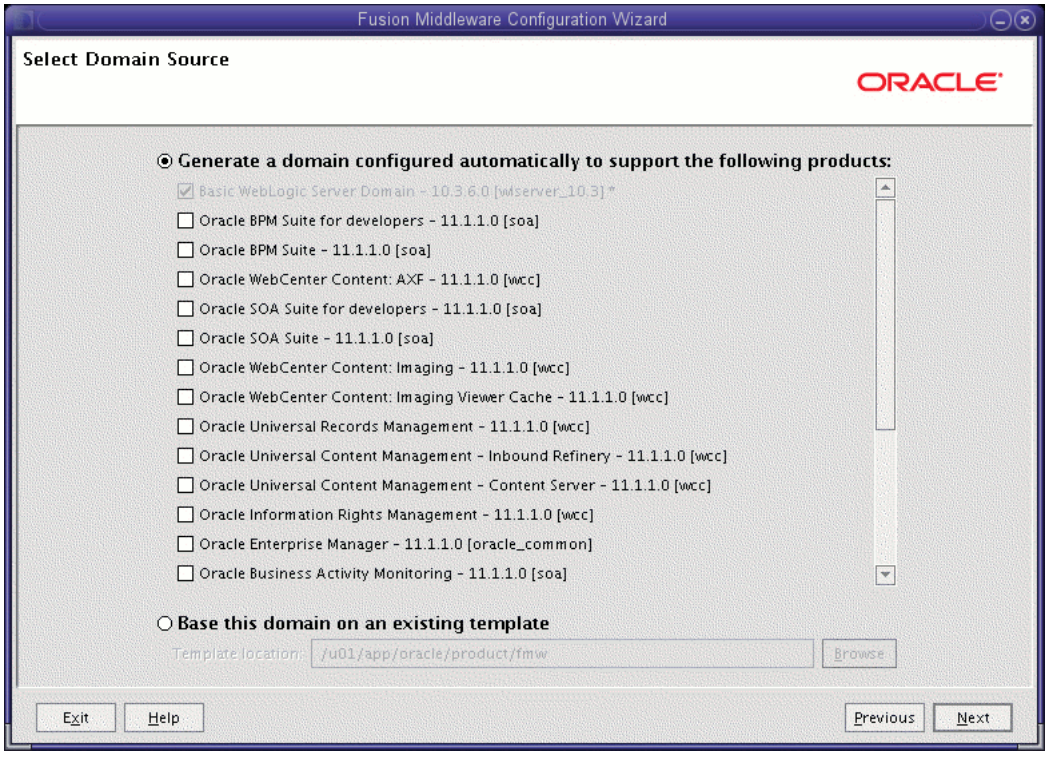
   ```
   cd ORACLE_COMMON_HOME/common/bin
   ```

3. Start the Configuration Wizard:

   ```
   ./config.sh
   ```

4. In the Welcome screen, select **Create a new WebLogic Domain**, and click **Next**.
5. The Select Domain Source screen opens (Figure 8–1).

Figure 8–1 Select Domain Source Screen



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 - 10.3.6.0 [wlserver_10.3]** (this should be selected automatically)
  - **Oracle Enterprise Manager - 11.1.1.0 [oracle_common]**

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

- **Oracle Enterprise Manager - 11.1.1.0 [oracle_common]**
- **Oracle JRF - 11.1.1.0 [oracle_common]**

Click **Next**.

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

   Make sure that the domain directory matches the directory and shared storage mount point recommended in Chapter 4, "Preparing the File System for an Enterprise Deployment." Enter *ORACLE_BASE*/admin/*domain_name*/aserver for the domain directory and *ORACLE_BASE*/admin/*domain_name*/aserver/applications for the application directory. This directory should be in shared storage.

   Click **Next**.

7. In the Configure Administrator User Name and Password screen, enter the user name and password to be used for the domain's administrator.

   Click **Next**.

8. In the Configure Server Start Mode and JDK screen, do the following:
   - For WebLogic Domain Startup Mode, select **Production Mode**.
   - For JDK Selection, select **JROCKIT SDK1.6.0_*version***.

   Click **Next**.

9. In the Select Optional Configuration screen, select the following:
   - **Administration Server**
   - **Managed Servers, Clusters and Machines**

   Click **Next**.

10. In the Configure the Administration Server screen, enter the following values:
    - **Name:** AdminServer
    - **Listen address:** ADMINVHN.
    - **Listen port:** 7001
    - **SSL listen port:** N/A
    - **SSL enabled:** Leave this checkbox unselected.

    Click **Next**.

11. In the Configure Managed Servers screen, click **Next**.

12. In the Configure Clusters screen, click **Next**.
13. In the Configure Machines screen, open the **Unix Machine** tab and then click **Add** to add the following machine:

Table 8–3 Machines

Name	Node Manager Listen Address
ADMINVHN	localhost

Leave all other fields set to their default values. The machine name does not need to be a valid host name or listen address; it is just a unique identifier of a Node Manager location.

> **Note:** The virtual host machine must point to `localhost` because `localhost` is the relative internal address for whatever machine is active. The Node Manager instance associated with the Administration Server changes when the Administration Server fails over because the Administration Server uses the `localhost` attribute in conjunction with the first host and then again, after failover, in conjunction with the second host.

Click **Next**.

14. In the Assign Servers to Machines screen, assign The Administration Server to a machine as follows:
    - **ADMINVHN:**
      - AdminServer

    Click **Next**.
15. In the Configuration Summary screen, click **Create**.
16. In the Create Domain screen, click **Done**.

8.4 Post-Configuration and Verification Tasks

After configuring the domain with the Configuration Wizard, follow these instructions for post-configuration and verification.

The section includes the following topics:

8.4.1 Creating boot.properties for the Administration Server on SOAHOST1

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

1. Create the following directory structure on SOAHOST1:

   ```
   mkdir -p ORACLE_BASE/admin/domain_name/aserver/domain_name/servers/
   AdminServer/security
   ```

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

   ```
   username=Admin_Username
   password=Password
   ```

   Note: When you start the Administration Server, the user name and password entries in the file get encrypted. You start the Administration Server in Section 8.4.3, "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.

3. Save the file, and close the editor.

8.4.2 Starting Node Manager on SOAHOST1

To start Node Manager on SOAHOST1, set the `StartScriptEnabled` property to `true`, and then start Node Manager using `startNodeManager.sh`.

To start Node Manager on SOAHOST1:

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:

   ```
   cd ORACLE_COMMON_HOME/common/bin

   ./setNMProps.sh
   ```

   Note: You must use the `StartScriptEnabled` property to avoid class loading failures and other problems. See also Section 16.12.3, "Incomplete Policy Migration After Failed Restart of SOA Server."

2. Start Node Manager:

   ```
   cd WL_HOME/server/bin

   export JAVA_OPTIONS=-DDomainRegistrationEnabled=true

   ./startNodeManager.sh
   ```

Note: It is important that the `-DDomainRegistrationEnabled=true` parameter is set whenever a Node Manager is started which must manage the Administration Server. If there is no Administration Server on the machine and the machine is not an Administration Server failover node, then Node Manager can be started using

```
./startNodeManager.sh
```

(without the `export` command).

8.4.3 Starting the Administration Server on SOAHOST1

The Administration Server is started and stopped using Node Manager. However, the first start of the Administration Server with Node Manager requires changing the default user name and password that the Oracle Fusion Middleware Configuration Wizard set for Node Manager. You must therefore use the start script for the Administration Server for the first start. Follow these steps to start the Administration Server using Node Manager (steps 1 through 4 are required for the first start; all subsequent starts require only step 4):

1. On SOAHOST1, start the Administration Server using the start script in the domain directory:

   ```
   cd ORACLE_BASE/admin/domain_name/aserver/domain_name/bin

   ./startWebLogic.sh
   ```

2. Use the Administration Console to update the Node Manager credentials:
   a. Open a web browser and go to `http://ADMINVHN:7001/console`.
   b. Log in as the administrator.
   c. Click **Lock & Edit**.
   d. Click ***domain_name***, then **Security**, then **General**, and then expand the **Advanced** options at the bottom.
   e. Enter a new user name for Node Manager, or make a note of the existing one, and update the Node Manager password.
   f. Save and activate the changes.
3. Stop the Administration Server process, either by using Ctrl+C in the shell where it was started or by the standard process identification and kill commands in the operating system.
4. Start the Oracle WebLogic Scripting Tool (WLST), connect to Node Manager with `nmconnect` and the credentials set in the previous steps, and start the Administration Server using `nmstart`:

   ```
   cd ORACLE_COMMON_HOME/common/bin

   ./wlst.sh
   ```

Once in the WLST shell, execute the following commands (make sure Node Manager is up and running):

```
wls:/offline>nmConnect("Admin_User","Admin_Pasword", "SOAHOST1","5556",
"domain_name","/u01/app/oracle/admin/domain_name/aserver/domain_name")

wls:/nm/domain_name> nmStart("AdminServer")
```

Note: SOAHOST1 is the address of the node where the domain was created, not the listen address of the Administration Server. Also, the user name and password are only used to authenticate connections between Node Manager and clients. They are independent from the server admin ID and password, and are stored in the *ORACLE_BASE*/admin/*domain_name*/aserver/*domain_name*/config/nodemanager/nm_password.properties file.

8.4.4 Validating the Administration Server Configuration

To ensure that the Administration Server for the domain you have created is properly configured, validate the configuration by logging in to the WebLogic Server Administration Console and verifying the Administration Server is listed, and then log in to Oracle Enterprise Manager Fusion Middleware Control.

To verify that the Administration Server is properly configured:

1. Open a Web browser and go to `http://ADMINVHN:7001/console`.
2. Log in as the administrator.
3. Check that you can access Fusion Middleware Control at `http://ADMINVHN:7001/em`.
4. Log in to Fusion Middleware Control with the user name and password you specified in Section 8.4.1, "Creating boot.properties for the Administration Server on SOAHOST1."

The Administration Server should be up and running.

8.4.5 Disabling Host Name Verification

This step is required if you have not set up the appropriate certificates to authenticate the different nodes with the Administration Server (see Chapter 13, "Setting Up Node Manager"). If you have not configured the server certificates, you will receive errors when managing the different WebLogic Server instances. To avoid these errors, disable host name verification for the Administration Server and each Managed Server while setting up and validating the topology, and enable it again once the enterprise deployment topology configuration is complete, as described in Chapter 13, "Setting Up Node Manager."

You can disable host name verification for a server as soon as it is up and running, starting with the Administration Server.

To disable host name verification:

1. Log in to the WebLogic Server Administration Console.
2. Click **Lock & Edit**.
3. In the **Domain Structure** tree on the left, expand the **Environment** node
4. Click **Servers**.

5. On the Summary of Servers page, click the name of a server in the **Names** column of the **Servers** table.
6. On the settings page for the server, click the **SSL** tab.
7. Expand the **Advanced** section of the page.
8. Set **Hostname Verification** to `None`.
9. Click **Save**.
10. Save and activate the changes.
11. This change requires a restart of Node Manager and the Administration Server:
    a. Stop the Administration Server with Node Manager, using the following command:

    ```
    wls:/nm/domain_name>nmKill("AdminServer")
    ```

    b. Stop Node Manager by stopping the process associated with it.

    If it is running in the foreground in a shell, use Ctrl+C.

    If it is running in the background in the shell, find the associated process and use the `kill` command to stop it. For example:

    ```
    ps -ef | grep NodeManager
    orcl   9139  9120  0 Mar03 pts/6    00:00:00 /bin/sh ./startNodeManager.sh

    kill -9 9139
    ```

    c. Start Node Manager:

    ```
    WL_HOME/server/bin/startNodeManager.sh
    ```

    d. Start the Administration Server again, as described in Section 8.4.3, "Starting the Administration Server on SOAHOST1."

8.5 Configuring Oracle HTTP Server for the WebLogic Server Domain

This section describes tasks for configuring Oracle HTTP Server for the WebLogic Server domain, and for verifying the configuration.

This section includes the following topics:

- Section 8.5.1, "Configuring Oracle HTTP Server for the Administration Server"
- Section 8.5.2, "Turning On the WebLogic Server Plug-In Enabled Flag"
- Section 8.5.3, "Registering Oracle HTTP Server with WebLogic Server"
- Section 8.5.4, "Setting the Front-End URL for the Administration Console and Setting Redirection Preferences"
- Section 8.5.5, "Validating Access Through the Load Balancer"
- Section 8.5.6, "Verifying Manual Failover of the Administration Server"

8.5.1 Configuring Oracle HTTP Server for the Administration Server

To enable Oracle HTTP Server to route to the Administration Server, you must set the corresponding mount points in your HTTP server configuration:

1. For each of the web servers on WEBHOST1 and WEBHOST2, add the following lines to the *ORACLE_INSTANCE*/config/OHS/ohs1/moduleconf/admin_vh.conf and *ORACLE_INSTANCE*/config/OHS/ohs2/moduleconf/admin_vh.conf files:

```
# Admin Server and EM
<Location /console>
   SetHandler weblogic-handler
   WebLogicHost ADMINVHN
   WebLogicPort 7001
   WLProxySSL OFF
   WLProxySSLPassThrough OFF
</Location>

<Location /consolehelp>
   SetHandler weblogic-handler
   WebLogicHost ADMINVHN
   WebLogicPort 7001
   WLProxySSL OFF
   WLProxySSLPassThrough OFF
</Location>

<Location /em>
   SetHandler weblogic-handler
   WebLogicHost ADMINVHN
   WebLogicPort 7001
   WLProxySSL OFF
   WLProxySSLPassThrough OFF
</Location>
```

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

```
ORACLE_BASE/admin/instance_name/bin/opmnctl restartproc ias-component=ohsX
```

   For WEBHOST1, use `ohs1` for `ias-component` and for WEBHOST2 use `ohs2`.

8.5.2 Turning On the WebLogic Server Plug-In Enabled Flag

For security purposes, and since the load balancer terminates SSL requests (Oracle HTTP Server routes the requests as non-SSL to WebLogic Server), once you configure SSL for the load balancer, turn on the WebLogic Server plug-in enabled flag for the domain.

To turn on the WebLogic Server plug-in enabled flag:

1. Log in to the Administration Console.
2. Click the domain name in the navigation tree on the left.
3. Open the **Web Applications** tab.
4. Click **Lock & Edit**.
5. Select the **WebLogic Plugin Enabled** checkbox.

6. Save and activate the changes.
7. Restart the Administration Server (even if the Administration Console does not specifically prompt for that).

8.5.3 Registering Oracle HTTP Server with WebLogic Server

Once a WebLogic Server domain is created, the Oracle Web Tier can be linked to the domain. The advantage of doing this is that the Oracle Web Tier can be managed and monitored using Oracle Enterprise Manager Fusion Middleware Control.

To associate the Oracle Web Tier with the WebLogic Server domain, run the following commands on WEBHOST1:

```
cd ORACLE_BASE/admin/instance_name/bin

./opmnctl registerinstance -adminHost ADMINVHN -adminPort 7001 -adminUsername
weblogic
```

You must also run this command from WEBHOST2 for ohs2.

After you register Oracle HTTP Server, it should appear as a manageable target in Fusion Middleware Control. To verify this, log in to Fusion Middleware Control. The WebTier item in the navigation tree should show that Oracle HTTP Server has been registered.

8.5.4 Setting the Front-End URL for the Administration Console and Setting Redirection Preferences

When you access the WebLogic Server Administration Console using a load balancer, it is required to change the Administration Server's front-end URL so that the user's web browser is redirected to the appropriate load balancer address.

The WebLogic Server Administration Console application tracks changes made to ports, channels and security using the console. When changes made through the console are activated, the console validates its current listen address, port and protocol. If the listen address, port and protocol are still valid, the console redirects the HTTP request, replacing the host and port information with the Administration Server's listen address and port.

To change the Administration Server's front-end URL:

1. Log in to the WebLogic Server Administration Console.
2. Click **Lock & Edit**.
3. Expand the **Environment** node in the **Domain Structure** tree on the left.
4. Click **Servers**.
5. On the Summary of Servers page, select **Admin Server** in the **Names** column of the table.
6. On the settings page for `AdminServer(admin)`, open the **Protocols** tab.
7. Open the **HTTP** tab.
8. Set the **Frontend Host** field to `admin.mycompany.com` and the **Frontend HTTP Port** field to 80 (modify accordingly if HTTPS is used for the admin URL).

9. Save and activate the changes.
10. Disable tracking on configuration changes in the WebLogic Server Administration Console so that the console does not trigger the reloading of configuration pages when activation of changes occurs.
    a. Log in to the WebLogic Server Administration Console.
    b. Click the **Preferences** link in the banner.
    c. Open the **Shared Preferences** tab.
    d. Clear the **Follow Configuration Changes** checkbox.
    e. Click **Save**.

Note: If you have any issues with activating any configuration changes after modifying the front-end host and port settings, see Section 16.12.10, "Redirection of Users to Login Screen After Activating Changes in the Administration Console."

8.5.5 Validating Access Through the Load Balancer

Verify that the server status is reported as `Running` in the Administration 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. For possible causes, see Section 16.12, "Troubleshooting the Oracle WebCenter Content Enterprise Deployment Topology."

Validate access to the Administration Console and Oracle Enterprise Manager Fusion Middleware Control through the load balancer using the following URLs:

- `http://admin.mycompany.com/console`
- `http://admin.mycompany.com/em`

For information on configuring system access through the load balancer, see Section 3.3, "Configuring the Load Balancers."

After you register Oracle HTTP Server as described in Section 8.5.3, "Registering Oracle HTTP Server with WebLogic Server," it should appear as a manageable target in Fusion Middleware Control. To verify this, log in to Fusion Middleware Control. The WebTier item in the navigation tree should show that Oracle HTTP Server has been registered.

8.5.6 Verifying Manual Failover of the Administration Server

After configuring the domain, test failover by following the steps in Section 16.8, "Verifying Manual Failover of the Administration Server."

8.6 Backing Up the WebLogic Server Domain Configuration

After you have verified that the extended domain is working, perform a backup to save your domain configuration. 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 that 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 that guide. For information on how to recover components, see the "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" section in the guide. For information about database backup, see the *Oracle Database Backup and Recovery User's Guide*.

To back up the installation at this point:

1. Back up the web tier from WEBHOST1:
   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 Oracle instance on the web tier using the following command:

      ```
      tar -cvpf BACKUP_LOCATION/web_instance_name.tar ORACLE_INSTANCE
      ```

   d. Start the instance using `opmnctl`:

      ```
      cd ORACLE_BASE/admin/instance_name/bin

      opmnctl startall
      ```

2. Repeat step 1 for WEBHOST2.
3. Back up the database. This is a full database backup (either hot or cold) using Oracle Recovery Manager (recommended) or operating system tools such as tar for cold backups if possible.
4. Stop the Administration Server and back up its domain directory to save your domain configuration. The configuration files all exist in the *ORACLE_BASE*`/admin/`*domain_name*`/` directory. Run the following command on SOAHOST1 to create the backup:

   ```
   tar -cvpf edgdomainback.tar ORACLE_BASE/admin/domain_name
   ```

   Remember to restart the Administration Server again.

9

Extending the Domain to Include Oracle SOA Suite Components

This chapter describes how to use the Oracle Fusion Middleware Configuration Wizard to extend the domain created in Chapter 8, "Creating a Domain for an Enterprise Deployment," to include Oracle SOA Suite components and Oracle WSM Policy Manager. You can extend the resulting domain to add other Fusion Middleware components, as described in the next chapters. It is assumed that a SOA Oracle home (binaries) has already been installed and is available from SOAHOST1 and SOAHOST2 and that a domain with an Administration Server has been created. This is the domain that will be extended in this chapter to support Oracle SOA Suite components.

> **Note:** Before starting the setup process, read the *Oracle Fusion Middleware Release Notes* for your platform for additional installation and deployment information.

This chapter contains the following sections:

9.1 Overview of Extending the Domain to Include Oracle SOA Suite Components

Extend the Oracle WebLogic Server domain to include Oracle SOA Suite components. Table 9–1 lists the steps for configuring Oracle SOA Suite and other tasks required for extending the domain for Oracle SOA Suite components.

Table 9–1 Steps for Extending the Domain for Oracle SOA Suite Components

Step	Description	More Information
Prepare for extending the domain for Oracle SOA Suite components	Enable a VIP mapping for each of the host names, and synchronize the system clocks for the Oracle SOA Suite WebLogic Server cluster	Section 9.2, "Enabling SOAHOST1VHN1 on SOAHOST1 and SOAHOST2VHN1 on SOAHOST2"
Extend the domain for Oracle SOA Suite components	Extend the WebLogic Server domain you created in Chapter 8, "Creating a Domain for an Enterprise Deployment."	Section 9.3, "Extending the Domain for Oracle SOA Suite Components"
Configure Oracle Coherence for deploying composites	Configure Oracle Coherence to use unicast communication for deploying composites.	Section 9.4, "Configuring Oracle Coherence for Deploying Composites"
Complete post-configuration and verification tasks	Follow these instructions for disabling host name verification, restarting Node Manager, propagating the domain configuration, starting and validating the Managed Servers, and validating GridLink data sources.	Section 9.5, "Completing Post-Configuration and Verification Tasks"
Configure the Java Object Cache (JOC) among all the servers running Oracle Web Services Manager (Oracle WSM)	Configure a local cache to increase the performance of Oracle WSM.	Section 9.6, "Configuring the Java Object Cache for Oracle Web Services Manager"
Configure Oracle HTTP Server with the extended domain	Configure the Oracle HTTP Server with the Managed Servers, validate access, set the front-end HTTP host and port, and set the WebLogic Server cluster address for SOA_Cluster.	Section 9.7, "Configuring Oracle HTTP Server with the Extended Domain"
Configure a default persistence store	Configure a default persistence store for transaction recovery.	Section 9.8, "Configuring a Default Persistence Store for Transaction Recovery"
Configure Oracle Adapters	Enable high availability for Oracle File and FTP Adapters and configure the Oracle Database Adapter.	Section 9.9, "Configuring Oracle Adapters"
Configure host name verification for the communication between Node Manager and the Managed Servers in the domain	Use certificates for the different addresses communicating with the Administration Server and other servers.	Section 9.10, "Configuring Node Manager for the WLS_SOA Managed Servers"
Configure server migration for the Oracle SOA Suite Managed Servers.	Specify the Oracle SOA Suite Managed Server names, host names, and cluster name for migration.	Section 9.11, "Configuring Server Migration for the WLS_SOA Managed Servers"
Back up the Oracle SOA Suite configuration	Back up the newly extended domain configuration.	Section 9.12, "Backing Up the Installation"

9.2 Enabling SOAHOST1VHN1 on SOAHOST1 and SOAHOST2VHN1 on SOAHOST2

The Oracle WebCenter Content domain uses virtual host names as the listen addresses for the Oracle SOA Suite Managed Servers. If you have not previously done so, you must enable a virtual IP mapping for each of these host names on the two SOAHOST machines, (VIP2 on SOAHOST1 and VIP3 on SOAHOST2), and correctly resolve the virtual host names in the network system used by the topology (either by DNS Server or `/etc/hosts` resolution).

To enable the virtual IPs, follow the procedure described in Section 3.5, "Enabling Virtual IP Addresses for an Enterprise Deployment" if you have not yet completed it. These virtual IPs and virtual host names are required to enable server migration for the Oracle SOA Suite servers. You can configure server migration for the Oracle SOA Suite servers later for high availability purposes. For more information about configuring server migration, see Chapter 14, "Configuring Server Migration for an Enterprise Deployment."

9.3 Extending the Domain for Oracle SOA Suite Components

Use the Configuration Wizard to extend the domain created in Chapter 8, "Creating a Domain for an Enterprise Deployment" to support Oracle Web Services Manager and Oracle SOA Suite components.

Note: If you have not backed up the domain created in Chapter 8, "Creating a Domain for an Enterprise Deployment," back up the current domain before extending it for Oracle SOA Suite components. You may use the backup to recover in case any errors are made in the domain extension. See "Backing Up Your Environment" in the *Oracle Fusion Middleware Administrator's Guide*.

To extend the domain for Oracle SOA Suite components:

1. Ensure that the database where you installed the repository is running.

   For Oracle RAC databases, Oracle recommends that all instances are running, so that the validation check later on becomes more reliable.

2. Shut down all Managed Servers in the domain.

3. On SOAHOST1, change the directory to the location of the Oracle Fusion Middleware Configuration Wizard. This is within the Oracle Common home directory (domain extensions are run from the node where the Administration Server resides).

   ```
   cd ORACLE_COMMON_HOME/common/bin
   ```

4. Start the Configuration Wizard:

   ```
   ./config.sh
   ```

5. In the Welcome screen, select **Extend an Existing WebLogic Domain**, and click **Next**.

6. In the Select a WebLogic Domain Directory screen, select the WebLogic Server domain directory (*ORACLE_BASE*`/admin/`*domain_name*`/aserver/`*domain_name*), and click **Next**.

7. In the Select Extension Source screen, which Figure 9–1 shows, do the following:

   1. Select **Extend my domain automatically to support the following added products**.

   2. Select these products:

      Oracle SOA Suite - 11.1.1.0 [soa]

      Oracle WSM Policy Manager - 11.1.1.0 [oracle_common] (automatically selected with Oracle SOA Suite)

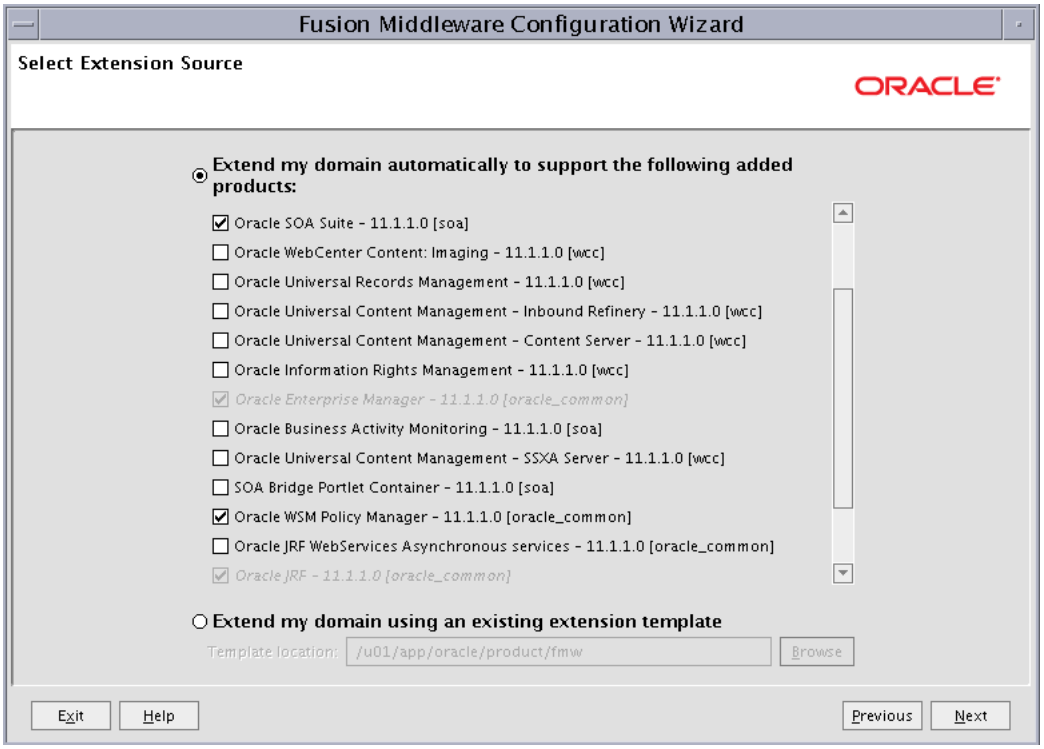
      The following products should already be selected and grayed out. They were selected when you created the domain (Section 8.3).

      Basic WebLogic Server Domain - 10.3.6.0 [wlserver_10.3]

      Oracle Enterprise Manager - 11.1.1.0 [oracle_common]

      Oracle JRF - 11.1.1.0 [oracle_common]

Figure 9–1 Select Extension Source Screen for Oracle SOA Suite



   3. Click **Next**.

8. 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**.

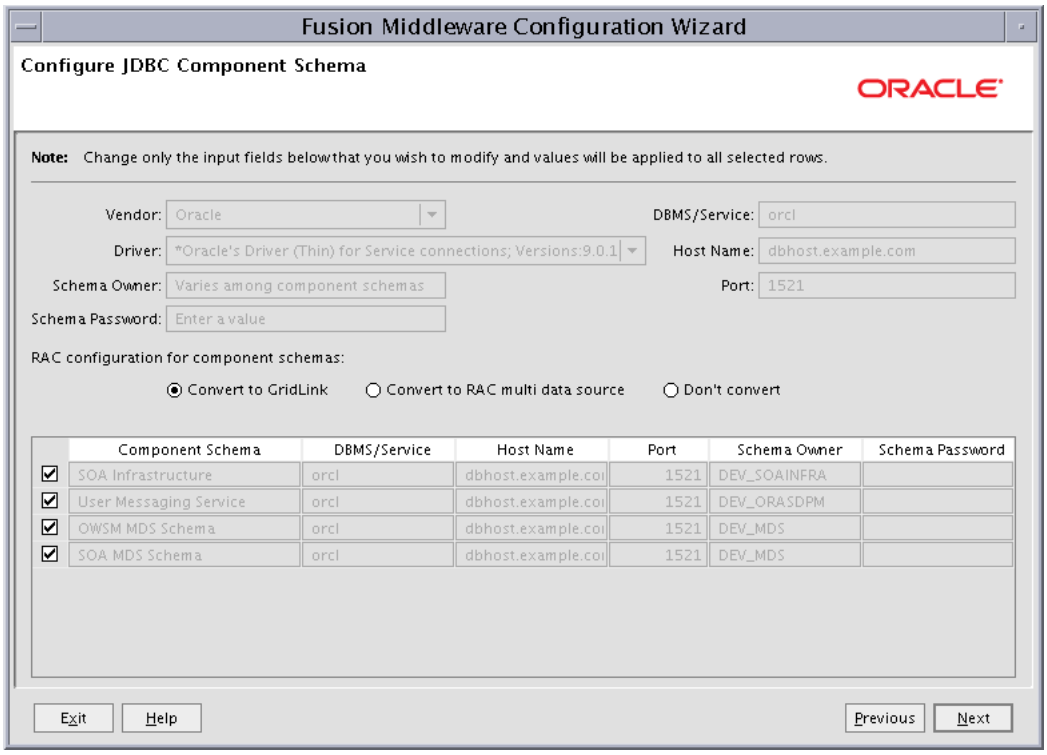
9. In the Configure JDBC Component Schema screen, which Figure 9–2 shows, do the following:

   a. Select the rows **SOA Infrastructure**, **User Messaging Service**, **OWSM MDS Schema**, and **SOA MDS Schema**.

   b. For the RAC configuration, you can select **Convert to GridLink** or **Convert to RAC multi data source** (described in Appendix A, "Using Multi Data Sources with Oracle RAC"). For the instructions given here, select **Convert to GridLink**.

      After you select a RAC configuration, all selected schemas are grayed.

Figure 9–2 Configure JDBC Component Schema Screen for Oracle SOA Suite

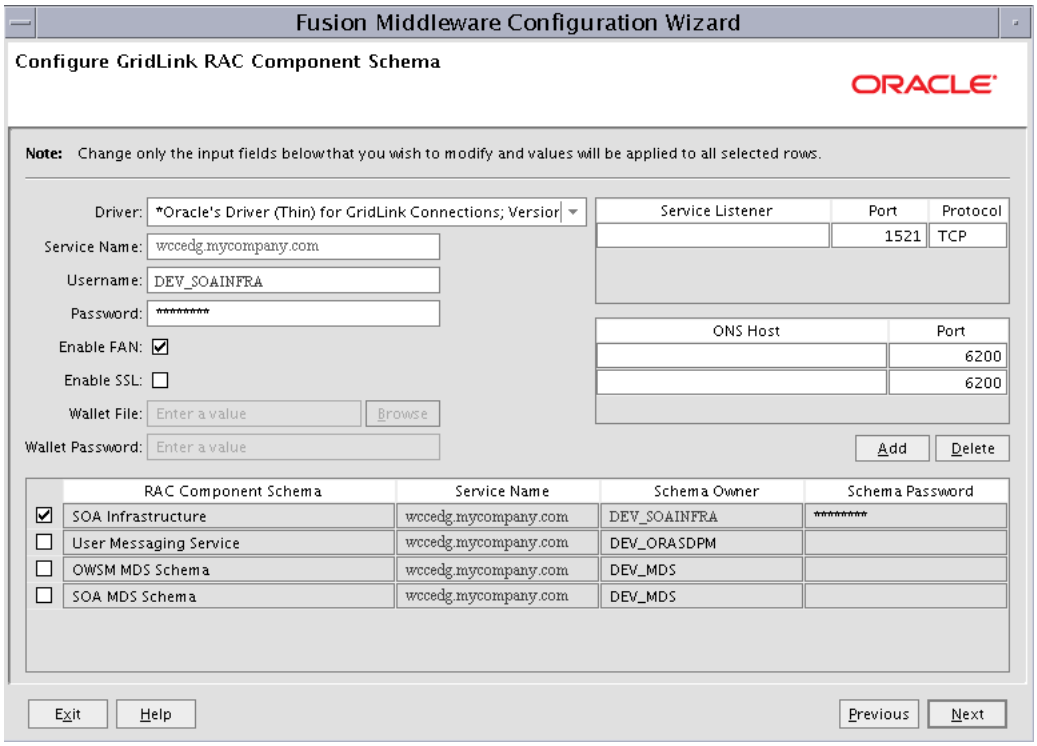


   c. Click **Next**.

10. In the Configure GridLink RAC Component Schema screen, which Figure 9–3 shows, do the following:

    a. Select only the **SOA Infrastructure** row.

Figure 9–3 Configure GridLink RAC Component Schema Screen for Oracle SOA Suite



    b. Enter values for the following fields, specifying the connection information for the GridLink RAC database that was seeded through RCU:

    - **Driver:** Select **Oracle driver (Thin) for GridLinkConnections; Versions:10 and later**.
    - **Service Name:** Enter the service name of the Oracle RAC database in lowercase letters, followed by the domain name; for example, `wccedg.mycompany.com`.
    - **Username:** Enter the complete user name for the database schema owner of the corresponding component.

      This book uses `DEV` as the prefix of user names for the database schemas.
    - **Password:** Enter the password for the database schema owner.
    - Select **Enable FAN**.
    - **Enable SSL:** Leave this option deselected.

      If you select SSL to enable Oracle Notification Service (ONS) notification encryption, provide the appropriate **Wallet File** and **Wallet Password** details.
    - **Service listener:** Enter the Oracle Single Client Access Name (SCAN) address and port for the Oracle RAC database being used. The protocol should be `TCP`.

Oracle recommends that you use a SCAN address to specify the Service Listener (and OSN Host) so you do not need to update a GridLink data source containing a SCAN address if you add or remove Oracle RAC nodes. To determine the SCAN address, query the `remote_listener` parameter in the database:

```
SQL>show parameter remote_listener;

NAME              TYPE        VALUE
-----             ------      -------
remote_listener   string      db-scan.mycompany.com:1521
```

Note: For Oracle Database 11*g* Release 1 (11.1), use the virtual IP and port of each database instance listener; for example:

```
custdbhost1-vip.mycompany.com (port 1521)
```

and

```
custdbhost2-vip.mycompany.com (1521)
```

For Oracle Database 10*g*, use multi data sources to connect to an Oracle RAC database. For information about configuring multi data sources, see Appendix A, "Using Multi Data Sources with Oracle RAC."

– **ONS Host:** Enter here also the SCAN address for the RAC database and the ONS remote port, as reported by the database:

```
[orcl@CUSTDBHOST2 ~]$ srvctl config nodeapps -s
ONS exists: Local port 6100, remote port 6200, EM port 2016
```

Note: For Oracle Database 11*g* Release 1 (11.1), use the host name and port of each database's ONS service; for example:

```
custdbhost1.mycompany.com (port 6200)
```

and

```
custdbhost2.mycompany.com (6200)
```

---

c. Select each data source, one at a time, and enter appropriate values for the user names and passwords.

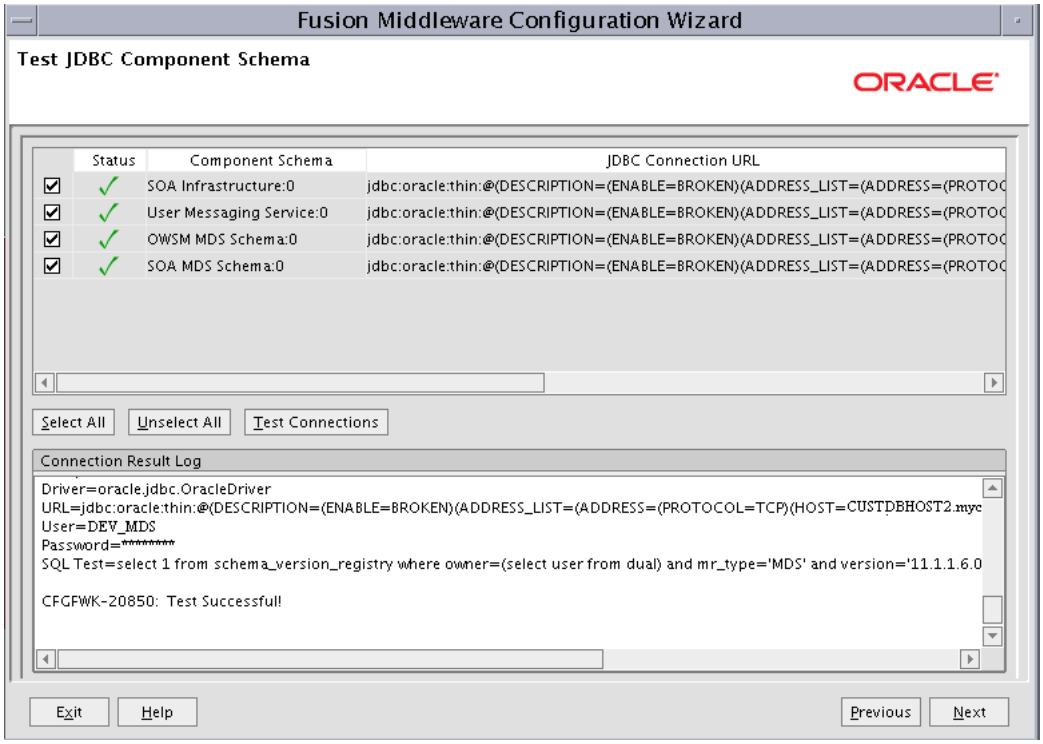
d. Ensure that all values are correct for all schemas (**SOA Infrastructure**, **User Messaging Service**, **OWSM MDS Schema**, and **SOA MDS Schema**), and then click **Next**.

Note: Oracle recommends using the database used for Oracle Identity Management (see Chapter 15, "Integrating with Oracle Identity Management") to store the Oracle WSM policies. It is therefore expected that you will use the Oracle Identity Management database information that has been seeded through RCU (as recommended in Chapter 3, "Preparing the Network for an Enterprise Deployment") to store the WSM metadata and that this Oracle Identity Management database information will be used in this screen for the OWSM MDS schemas. (This database connection information will be different from the one used for the rest of the Oracle SOA Suite schemas.)

11. In the Test JDBC Component Schema screen, (Figure 9–4), select each schema, and then click **Test Connections**.

    The **Connection Results Log** displays the results. Ensure that all connections were successful. If not, click **Previous** to return to the previous screen, correct your entries, and then retry the test.

Figure 9–4 Test JDBC Component Schema Screen



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

12. In the Select Optional Configuration screen, select the following options:
    - **JMS Distributed Destination**
    - **Managed Servers, Clusters and Machines**
    - **Deployments and Services**
    - **JMS File Store**

    Click **Next**.
13. In the Select JMS Distributed Destination Type screen, do the following:
    - Select **UDD** from the drop-down list for **UMSJMSystemResource**.
    - Select **UDD** from the drop-down list for **SOAJMSModule**.
    - Select **UDD** from the drop-down list for **BPMJMSModule**.

    If an override warning appears, click **OK** to acknowledge it.
14. 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 add a new server called WLS_SOA2. Give these servers the attributes listed in Table 9–2. Do not modify the other servers that are shown in this screen; leave them as they are.

Table 9–2 Managed Servers

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled
WLS_SOA1	SOAHOST1VHN1	8001	n/a	No
WLS_SOA2	SOAHOST2VHN1	8001	n/a	No

    Click **Next**.
15. In the Configure Clusters screen, add the clusters as listed in Table 9–3.

Table 9–3 Clusters

Name	Cluster Messaging Mode	Multicast Address	Multicast Port	Cluster Address
SOA_Cluster	unicast	n/a	n/a	SOAHOST1VHN1:8001,SOAHOST2VHN1:8001

    Click **Next**.

Note: For asynch request/response interactions over direct binding, the SOA composites must provide their JNDI provider URL for the invoked service to look up the beans for callbacks.

If soa-infra config properties are not specified, but the WebLogic Server cluster address is specified, the cluster address from the JNDI provider URL is used. This cluster address can be either a single DNS name that maps to the IP addresses of the clustered servers or a comma-separated list of *`server ip:port`*. Alternatively, the soa-infra config property `JndiProviderURL/SecureJndiProviderURL` can be used for the same purpose if explicitly set by users.

---

16. In the Assign Servers to Clusters screen, assign servers to clusters as follows:
    - **SOA_Cluster:**
      - WLS_SOA1
      - WLS_SOA2

    Click **Next**.

17. In the Configure Machines screen, delete the LocalMachine that appears by default and open the **Unix Machine** tab. You should add the SOAHOST1 and SOAHOST2 machines and eventually have the following entries:

Table 9–4 Machines

Name	Node Manager Listen Address
SOAHOST1	SOAHOST1
SOAHOST2	SOAHOST2
ADMINVHN	localhost

    Do not modify the other assignments that are shown in this screen; leave them as they are. Please note that the machine names do not need to be valid host names or listen addresses; they are just unique identifiers of Node Manager locations.

    Click **Next**.

18. In the Assign Servers to Machines screen, assign servers to machines as follows:
    - **ADMINVHN:**
      - AdminServer
    - **SOAHOST1:**
      - WLS_SOA1
    - **SOAHOST2:**
      - WLS_SOA2

    Click **Next**.

19. 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.)
    - **WSM-PM** should be targeted only to **SOA_Cluster**.
    - The **oracle.rules#***, **oracle.sdp.***, and **oracle.soa.*** deployments should be targeted only to **SOA_Cluster**.

    Click **Next**.

20. In the Target Services to Clusters or Servers screen, click **Next**.

21. In the Configure JMS File Stores screen, enter the shared directory location specified for your JMS stores, as recommended in Section 4.3, "About Recommended Locations for the Different Directories." For example:

    `ORACLE_BASE/admin/domain_name/soa_cluster_name/jms`

    Click **Next**.

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

    The domain is extended to include the Oracle SOA Suite components.

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

24. Restart the Administration Server for these changes to take effect.

    To restart the Administration Server, stop it first using the Administration Console and then start it again as described in Section 8.4.3, "Starting the Administration Server on SOAHOST1."

9.4 Configuring Oracle Coherence for Deploying Composites

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

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 Oracle SOA Suite enterprise deployments where multiple IPs are available in the same system, you must configure Oracle Coherence to use a specific host name to create the Oracle Coherence cluster.

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

9.4.1 Enabling Communication for Deployment Using Unicast Communication

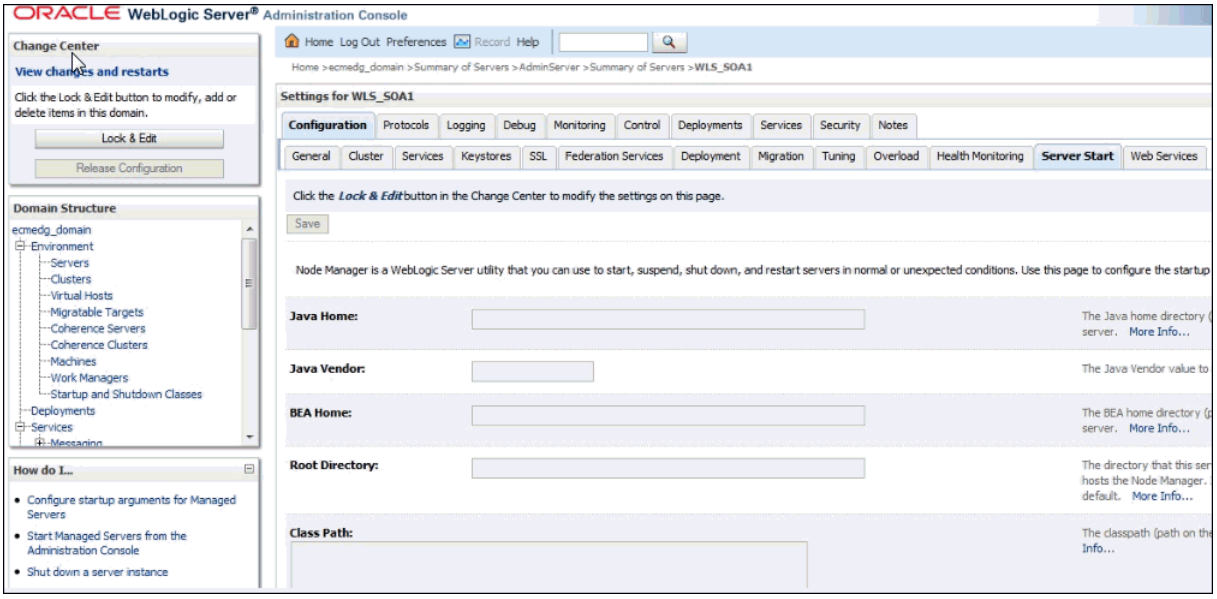
Specify the nodes using the `tangosol.coherence.wkan` system property, where *n* is a number in the range `1` to `9`. You can specify up to 9 nodes as Well Known Addresses, but the cluster can have more than 9 nodes. Start the numbering at 1. This numbering must be sequential and must not contain gaps.

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

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 Oracle SOA Suite server as the listener addresses (SOAHOST1VHN1 and SOAHOST2VHN1). Set this property by adding the `-Dtangosol.coherence.localhost` parameter to the **Arguments** field on the **Server Start** tab of the Oracle WebLogic Server Administration Console (Figure 9–5).

> **Note:** SOAHOST1VHN1 is the virtual host name that maps to the virtual IP where WLS_SOA1 is listening (in SOAHOST1). SOAHOST2VHN1 is the virtual host name that maps to the virtual IP where WLS_SOA2 is listening (in SOAHOST2).

Figure 9–5 Setting the Host Name on the Start Server Tab of the WebLogic Server Administration Console



9.4.2 Specifying the Host Name Used by Oracle Coherence

Use the Administration Console to add a host name for Oracle Coherence to use.

To specify the host name used by Oracle Coherence:

1. Log in to the WebLogic Server Administration Console.
2. In the **Domain Structure** tree on the left, expand the **Environment** node.
3. Click **Servers**.
4. On the Summary of Servers page, click the name of the server (**WLS_SOA1** or **WLS_SOA2**, which are represented as hyperlinks) in the **Name** column of the table.
5. On the settings page for the selected server, click **Lock & Edit**.
6. On the **Configuration** tab, click the **Server Start** tab (illustrated in Figure 9–5).
7. Enter the following parameters for WLS_SOA1 and WLS_SOA2 in the **Arguments** field.

Note: There should be no breaks in lines between the different `-D` parameters. Do not copy or paste the text to the **Arguments** text field in your Administration Console because this could result in HTML tags being inserted into the Java arguments. The text should not contain other text characters than those included the example that follows.

For WLS_SOA1, enter the following text (on a single line, without a carriage return):

```
-Dtangosol.coherence.wka1=SOAHOST1VHN1
-Dtangosol.coherence.wka2=SOAHOST2VHN1
-Dtangosol.coherence.localhost=SOAHOST1VHN1
```

For WLS_SOA2, enter the following text (on a single line, without a carriage return):

```
-Dtangosol.coherence.wka1=SOAHOST1VHN1
-Dtangosol.coherence.wka2=SOAHOST2VHN1
-Dtangosol.coherence.localhost=SOAHOST2VHN1
```

Note: The Coherence cluster used for deployment uses port 8088 by default. This port can be changed by specifying a different port (for example, 8089) using the `-Dtangosol.coherence.wkan.port` and `-Dtangosol.coherence.localport` startup parameters; for example:

- WLS_SOA1 (enter the following text into the **Arguments** field on a single line, without a carriage return):

  ```
  -Dtangosol.coherence.wka1=SOAHOST1VHN1
  -Dtangosol.coherence.wka2=SOAHOST2VHN1
  -Dtangosol.coherence.localhost=SOAHOST1VHN1
  -Dtangosol.coherence.localport=8089
  -Dtangosol.coherence.wka1.port=8089
  -Dtangosol.coherence.wka2.port=8089
  ```

- WLS_SOA2 (enter the following text into the **Arguments** field on a single line, without a carriage return):

  ```
  -Dtangosol.coherence.wka1=SOAHOST1VHN1
  -Dtangosol.coherence.wka2=SOAHOST2VHN1
  -Dtangosol.coherence.localhost=SOAHOST2VHN1
  -Dtangosol.coherence.localport=8089
  -Dtangosol.coherence.wka1.port=8089
  -Dtangosol.coherence.wka2.port=8089
  ```

For more information about Coherence clusters, see the *Oracle Coherence Developer's Guide*.

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

Notes:

- 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.
- The multicast and unicast addresses are different from the ones used by the WebLogic Server cluster for cluster communication. Oracle SOA Suite 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.

9.5 Completing Post-Configuration and Verification Tasks

After extending the domain with the Configuration Wizard and configuring Oracle Coherence, follow these instructions for post-configuration and validation.

This section includes the following topics:

9.5.1 Disabling Host Name Verification for the WLS_SOA*n* Managed Servers

For the enterprise deployment described in this guide, you set up the appropriate certificates to authenticate the different nodes with the Administration Server after you have completed the procedures to extend the domain for Oracle SOA Suite. You must disable the host name verification for the WLS_SOA1 and WLS_SOA2 Managed Servers to avoid errors when managing the different WebLogic Server instances. For more information, see Section 8.4.5, "Disabling Host Name Verification."

You enable host name verification again once the enterprise deployment topology configuration is complete. For more information, see Section 13.3, "Enabling Host Name Verification Certificates for Node Manager."

9.5.2 Propagating the Domain Changes to the Managed Server Domain Directory

Propagate the start scripts and classpath configuration from the Administration Server's domain directory to the Managed Server domain directory.

To propagate the start scripts and classpath configuration:

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:

```
cd ORACLE_COMMON_HOME/common/bin

./pack.sh -managed=true -domain=ORACLE_BASE/admin/domain_name/aserver/domain_
name -template=edgdomaintemplateSOA.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:

```
./unpack.sh -domain=ORACLE_BASE/admin/domain_name/mserver/domain_name
-overwrite_domain=true -template=edgdomaintemplateSOA.jar -app_dir=ORACLE_
BASE/admin/domain_name/mserver/applications
```

> **Note:** The `-overwrite_domain` option in the `unpack` command, allows unpacking a Managed Server template into an existing domain and existing applications directories. For any file that is overwritten, a backup copy of the original is created. If any modifications had been applied to the start scripts and EAR files in the Managed Server domain directory, they must be restored after this unpack operation.

9.5.3 Starting and Validating the WLS_SOA1 Managed Server

You can start and validate the WLS_SOA1 Managed Server through the Administration Console.

To start the WLS_SOA1 Managed Server on SOAHOST1:

1. Start the WLS_SOA1 Managed Server using the WebLogic Server Administration Console, as follows:
   a. Access the Administration Console at `http://ADMINVHN:7001/console`.

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

   b. Expand the **Environment** node in the **Domain Structure** tree on the left.
   c. Click **Servers**.
   d. On the Summary of Servers page, open the **Control** tab.
   e. Select **WLS_SOA1** and then click **Start**.

2. Verify that the server status is reported as `Running` in the Administration 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 16.12, "Troubleshooting the Oracle WebCenter Content Enterprise Deployment Topology" for possible causes.
3. Access the following URLs:
   - Access `http://SOAHOST1VHN1:8001/soa-infra` to verify the status of WLS_SOA1.
   - Access `http://SOAHOST1VHN1:8001/wsm-pm` to verify the status of Web Services Manager. Click **Validate Policy Manager**. A list of policies and assertion templates available in the data store opens.The configuration is incorrect if no policies or assertion templates appear.
   - Access `http://SOAHOST1VHN1:8001/integration/worklistapp` to verify the status of the worklist application.

9.5.4 Propagating the Domain Configuration to SOAHOST2

After completing the configuration of SOAHOST1, propagate the configuration to SOAHOST2 using the unpack utility, and then validate the propagated configuration.

To propagate the domain configuration:

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

```
cd ORACLE_BASE/product/fmw/oracle_common/common/bin

scp edgdomaintemplateSOA.jar oracle@SOAHOST2:ORACLE_BASE/product/fmw/oracle_
common/common/bin
```

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

> **Note:** Run `unpack` from the *`ORACLE_COMMON_HOME`*`/common/bin/` directory, not from *`WL_HOME`*`/common/bin/`.

```
cd ORACLE_COMMON_HOME/common/bin

./unpack.sh -domain=ORACLE_BASE/admin/domain_name/mserver/domain_name
-template=edgdomaintemplateSOA.jar -app_dir=ORACLE_BASE/admin/domain_
name/mserver/applications
```

> **Note:** The *`ORACLE_BASE`*`/admin/`*`domain_name`*`/mserver/` directory must exist before you run `unpack`. In addition, the *`ORACLE_BASE`*`/admin/`*`domain_name`*`/mserver/applications/` directory must be empty.

9.5.5 Starting and Validating the WLS_SOA2 Managed Server

After you start the WLS_SOA2 Managed Server, you can validate it by verifying its status through the WebLogic Server Administration Console and some URLs.

Note: Before starting the WLS_SOA2 Managed Server using the WebLogic Server Administration Console, make sure Node Manager is running. Section 16.8.1, "Assumptions and Procedure," says to start it, in Step 3, "Start Node Manager in SOAHOST2."

If Node Manager is not running, restart it on SOAHOST2, as described in Section 8.4.2, "Starting Node Manager on SOAHOST1."

To start and validate the WLS_SOA2 Managed Server:

1. Start the WLS_SOA2 Managed Server using the WebLogic Server Administration Console, as follows:
   a. Expand the Environment node in the **Domain Structure** tree on the left.
   b. Click **Servers**.
   c. On the Summary of Servers page, open the **Control** tab.
   d. Select **WLS_SOA2** and then click **Start**.
2. Verify that the server status is reported as `Running` in the Administration 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 16.12, "Troubleshooting the Oracle WebCenter Content Enterprise Deployment Topology" for possible causes.
3. Access the following URLs:
   - Access `http://SOAHOST2VHN1:8001/soa-infra` to verify the status of WLS_SOA2.
   - Access `http://SOAHOST2VHN1:8001/wsm-pm` to verify the status of Web Services Manager. Click **Validate Policy Manager**. A list of policies and assertion templates available in the data store opens.

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

   - Access `http://SOAHOST2VHN1:8001/integration/worklistapp` to verify the status of the worklist application.

9.5.6 Validating GridLink Data Sources for Oracle SOA Suite

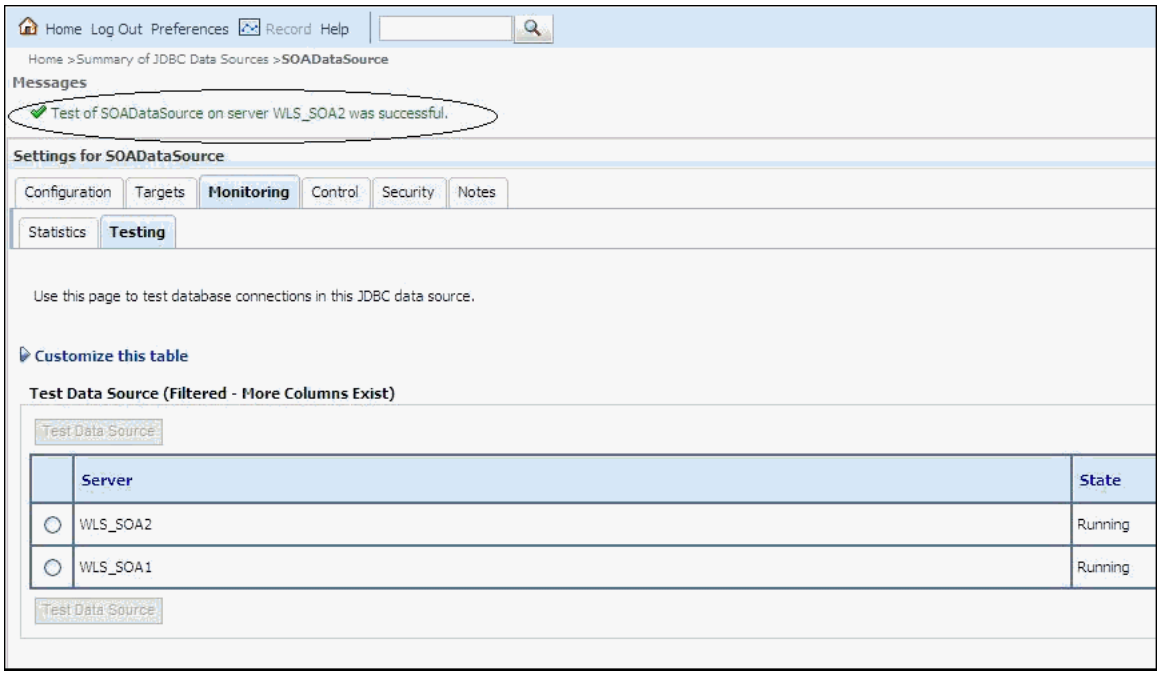
After the servers are started, verify that the GridLink data sources are correctly configured and that the ONS setup is correct. Perform this procedure for every GridLink data source created.

To verify the configuration of a GridLink data source for Oracle SOA Suite:

1. Log in to the WebLogic Server Administration Console.
2. In the **Domain Structure** tree, expand **Services**, then click **Data Sources**.
3. Click the name of a GridLink data source that was created.
4. Click the **Monitoring** tab.

5. Click the **Testing** tab (Figure 9–6), select one of the servers, and click **Test Data Source**.

Figure 9–6 Testing a GridLink Data Source for Oracle SOA Suite



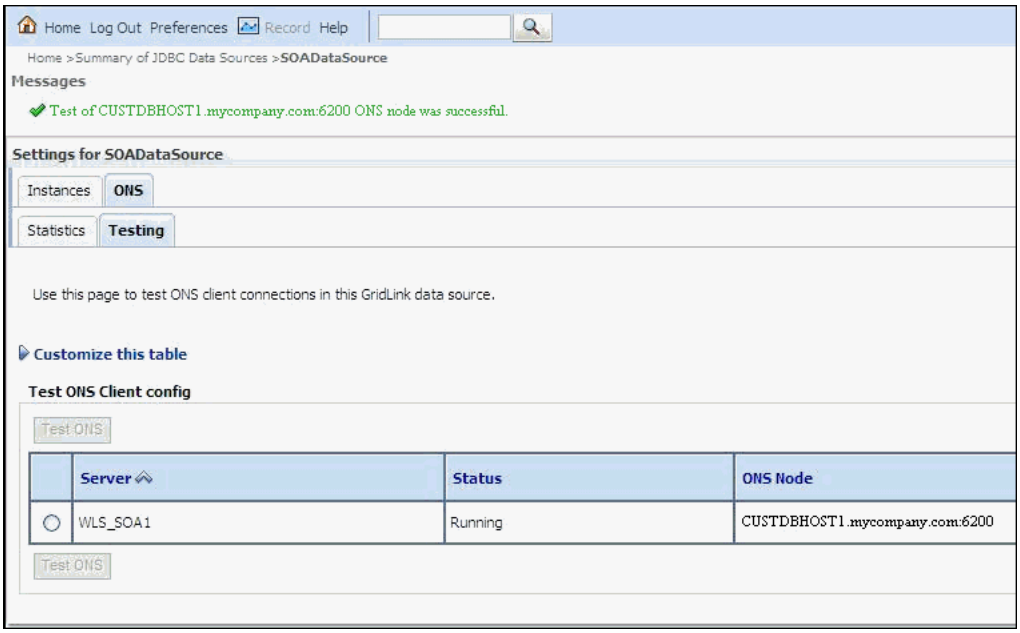
The test should be successful if the configuration is correct.

6. Repeat the test for every WebLogic Server instance that uses the GridLink data source.

To verify the configuration of ONS for a GridLink data source for Oracle SOA Suite:

1. Log in to the WebLogic Server Administration Console.
2. In the **Domain Structure** tree, expand **Services**, then click **Data Sources**.
3. Click the name of a GridLink data source that was created.
4. Click the **Monitoring** tab.
5. Click the **ONS** tab and then the **Testing** tab (Figure 9–7).
6. Select a server, and click **Test ONS**.

Figure 9–7 Testing the ONS Configuration for Oracle SOA Suite



The test should be successful if the configuration is correct. If the ONS test fails, verify that the ONS service is running in the Oracle RAC database nodes:

```
[orcl@CUSTDBHOST1 ~]$ srvctl status scan_listener
SCAN Listener LISTENER_SCAN1 is enabled
SCAN listener LISTENER_SCAN1 is running on node CUSTDBHOST1
SCAN Listener LISTENER_SCAN2 is enabled
SCAN listener LISTENER_SCAN2 is running on node CUSTDBHOST2
SCAN Listener LISTENER_SCAN3 is enabled
SCAN listener LISTENER_SCAN3 is running on node CUSTDBHOST2


[orcl@CUSTDBHOST1 ~]$ srvctl config nodeapps -s
ONS exists: Local port 6100, remote port 6200, EM port 2016


[orcl@CUSTDBHOST1 ~]$ srvctl status nodeapps | grep ONS
ONS is enabled
ONS daemon is running on node: CUSTDBHOST1
ONS daemon is running on node: CUSTDBHOST2
```

7. Repeat the ONS test for every WebLogic Server instance that uses the GridLink data source.

9.6 Configuring the Java Object Cache for Oracle Web Services Manager

The Java Object Cache (JOC) should be configured among all the servers running Oracle Web Services Manager (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.

For the configuration of JOC ports for Oracle products, Oracle recommends using ports in the 9988 to 9998 range.

Note: After you configure 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.

To configure the Java Object Cache for Oracle WSM:

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

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

   Enter the WebLogic Server Administrator user name, password, and server URL (`t3://ADMINVHN:7001`) when prompted.

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

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

Specifically, for enterprise deployment environments, the first cluster option in step 2 should be used. Here is a walkthrough for using `configure-joc.py` for EDG environments (see the following text for the script input parameters):

```
Execfile("MW_HOME/oracle_common/bin/configure-joc.py").
Enter Hostnames (eg host1,host2) : SOAHOST1VHN1,SOAHOST2VHN1
.
Do you want to specify a cluster name (y/n) <y>y
.
Enter Cluster Name : SOA_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
```

After configuring the Java Object Cache using the WLST commands or *configure-joc.py* script, restart all affected Managed Servers for the configurations to take effect.

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

Script Input Parameters

The input parameters are prompted by the script. The script can be used to perform the following optional JOC configurations:

- **Configure JOC for all specified 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 configures 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 : SOA_Cluster
  Enter Discover Port : 9991
  ```

- **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_SOA1:9991, WLS_SOA21:9991) :
WLS_SOA1:9999,WLS_SOA2:9999
```

  This example configures JOC only for the specified Managed Servers (that is, WLS_SOA1 and WLS_SOA2). The discover port is specified with the Managed Server (for example, `WLS_SOA1:2222`).

- **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_SOA1,WLS_SOA2
```

- **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`.

9.7 Configuring Oracle HTTP Server with the Extended Domain

After propagating the domain configuration to SOAHOST2, configure the Oracle HTTP Server with the extended domain.

This section includes the following topics:

- Section 9.7.1, "Configuring Oracle HTTP Server for the WLS_SOA Managed Servers"
- Section 9.7.2, "Setting the Front-End HTTP Host and Port"
- Section 9.7.3, "Validating Access Through the Load Balancer"

9.7.1 Configuring Oracle HTTP Server for the WLS_SOA Managed Servers

To enable Oracle HTTP Server to route to SOA_Cluster, which contain the WLS_SOA*n* Managed Servers, you must set the `WebLogicCluster` parameter to the list of nodes in the cluster:

1. For each of the web servers on WEBHOST1 and WEBHOST2, add the following lines to the *`ORACLE_INSTANCE`*`/config/OHS/ohs1/moduleconf/soainternal_vh.conf` and *`ORACLE_INSTANCE`*`/config/OHS/ohs2/moduleconf/soainternal_vh.conf` files:

```
# WSM-PM
<Location /wsm-pm>
    WebLogicCluster SOAHOST1VHN1:8001,SOAHOST2VHN1:8001
    SetHandler weblogic-handler
    WLProxySSL OFF
    WLProxySSLPassThrough OFF
</Location>

# SOA soa-infra app
```

```
<Location /soa-infra>
    WebLogicCluster SOAHOST1VHN1:8001,SOAHOST2VHN1:8001
    SetHandler weblogic-handler
    WLProxySSL OFF
    WLProxySSLPassThrough OFF
</Location>

# SOA inspection.wsil
<Location /inspection.wsil>
    WebLogicCluster SOAHOST1VHN1:8001,SOAHOST2VHN1:8001
    SetHandler weblogic-handler
    WLProxySSL OFF
    WLProxySSLPassThrough OFF
</Location>

# Worklist
<Location /integration/>
    WebLogicCluster SOAHOST1VHN1:8001,SOAHOST2VHN1:8001
    SetHandler weblogic-handler
    WLProxySSL OFF
    WLProxySSLPassThrough OFF
</Location>

# UMS prefs
<Location /sdpmessaging/userprefs-ui>
    WebLogicCluster SOAHOST1VHN1:8001,SOAHOST2VHN1:8001
    SetHandler weblogic-handler
    WLProxySSL OFF
    WLProxySSLPassThrough OFF
</Location>

# Default to-do taskflow
<Location /DefaultToDoTaskFlow/>
    WebLogicCluster SOAHOST1VHN1:8001,SOAHOST2VHN1:8001
    SetHandler weblogic-handler
    WLProxySSL OFF
    WLProxySSLPassThrough OFF
</Location>

# Workflow
<Location /workflow>
    WebLogicCluster SOAHOST1VHN1:8001,SOAHOST2VHN1:8001
    SetHandler weblogic-handler
    WLProxySSL OFF
    WLProxySSLPassThrough OFF
</Location>

#SOA Composer
<Location /soa/composer>
    WebLogicCluster SOAHOST1VHN1:8001,SOAHOST2VHN1:8001
    SetHandler weblogic-handler
    WLProxySSL OFF
    WLProxySSLPassThrough OFF
</Location>
```

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

   ORACLE_BASE/admin/*instance_name*/bin/opmnctl restartproc ias-component=ohsX

   For WEBHOST1, use `ohs1` for `ias-component` and for WEBHOST2 use `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. Please 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 run time.
- 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*.

9.7.2 Setting the Front-End HTTP Host and Port

You must set the front-end HTTP host and port for the Oracle WebLogic Server cluster hosting the Oracle SOA Suite servers:

1. Log in to the WebLogic Server Administration Console.
2. Go to the Change Center section and click **Lock & Edit**.
3. Expand the **Environment** node in the **Domain Structure** tree on the left.
4. Click **Clusters**.
5. On the Summary of Servers page, select **SOA_Cluster**.
6. Open the **HTTP** tab.
7. Set the following values:
   - **Frontend Host:** soainternal.mycompany.com
   - **Frontend HTTP Port:** 80
8. Click **Save**.
9. Click **Activate Changes** in the Change Center section of the Administration Console.
10. Restart the WLS_SOA1 and WLS_SOA2 Managed Servers to make the front-end host directive in the cluster take effect.

9.7.3 Validating Access Through the Load Balancer

Verify that the server status is reported as `Running` in the Administration 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. For possible causes, see Section 16.12, "Troubleshooting the Oracle WebCenter Content Enterprise Deployment Topology."

Validate access to `SOA_cluster` using the following URLs:

- `http://soainternal.mycompany.com/soa-infra`
- `http://soainternal.mycompany.com/integration/worklistapp`
- `http://soainternal.mycompany.com/sdpmessaging/userprefs-ui`
- `http://soainternal.mycompany.com/soa/composer`
- `http://soainternal.mycompany.com/wsm-pm`

For information on configuring system access through the load balancer, see Section 3.3, "Configuring the Load Balancers."

Note: After the registering Oracle HTTP Server as described in Section 8.5.3, "Registering Oracle HTTP Server with WebLogic Server," Oracle HTTP Server should appear as a manageable target in the Oracle Enterprise Manager console. To verify this, log in to the Oracle Enterprise Manager console. The WebTier item in the navigation tree should show that Oracle HTTP Server has been registered.

9.8 Configuring a Default Persistence Store for Transaction Recovery

Each server has a transaction log that stores information about the committed transactions coordinated by the server that may not have been completed. Oracle 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 Managed Servers within a cluster, store the transaction logs in a location accessible to each server and its backup servers.

Note: The recommended location is on a dual-ported SCSI disk or on a Storage Area Network (SAN).

To set the location for the default persistence stores:

1. Log in to the WebLogic Server Administration Console.
2. In the Change Center section of the page, click **Lock & Edit**.
3. In the **Domain Structure** tree on the left, expand the **Environment** node, and then click the **Servers** node.
4. On the Summary of Servers page, click the name of the server WLS_SOA1 (represented as a hyperlink) in the **Name** column of the table. The settings page for the selected server opens and defaults to the **Configuration** tab.
5. Click the **Configuration** tab and then the **Services** tab (not the top-level **Services** tab).
6. In the Default Store section of the page, enter the path to the folder where the default persistent store will store its data files. The directory structure of the path follows:

   ORACLE_BASE/admin/*domain_name*/*soa_cluster_name*/tlogs/

7. Repeat steps 4, 5, and 6 for the WLS_SOA2 server.
8. Click **Save** and **Activate Changes**.

9. Restart the WLS_SOA1 and WLS_SOA2 Managed Servers.
10. After WLS_SOA1 and WLS_SOA2 are restarted, verify that the following `DAT` files are created in this directory:

    ORACLE_BASE/admin/*domain_name*/*soa_cluster_name*/tlogs/

    - `_WLS_WLS_SOA1000000.DAT`
    - `_WLS_WLS_SOA2000000.DAT`

    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. Also, the directory must exist before you restart the servers.

9.9 Configuring Oracle Adapters

Configure Oracle File, FTP, and database adapters for the extended Oracle SOA Suite domain.

This section includes the following topics:

- Section 9.9.1, "Enabling High Availability for Oracle File and FTP Adapters"
- Section 9.9.2, "Configuring the Oracle Database Adapter"

9.9.1 Enabling High Availability for Oracle File and FTP Adapters

Note: This step is optional and applies only to those deployments that require adapter support for the BPEL processes that are invoked.

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 the *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 nontransactional, files can be processed twice. As a result, it is possible to get duplicate files when there is failover in the Oracle RAC back end or in the Oracle SOA Suite Managed Servers.

Using the Database Mutex Locking Operation

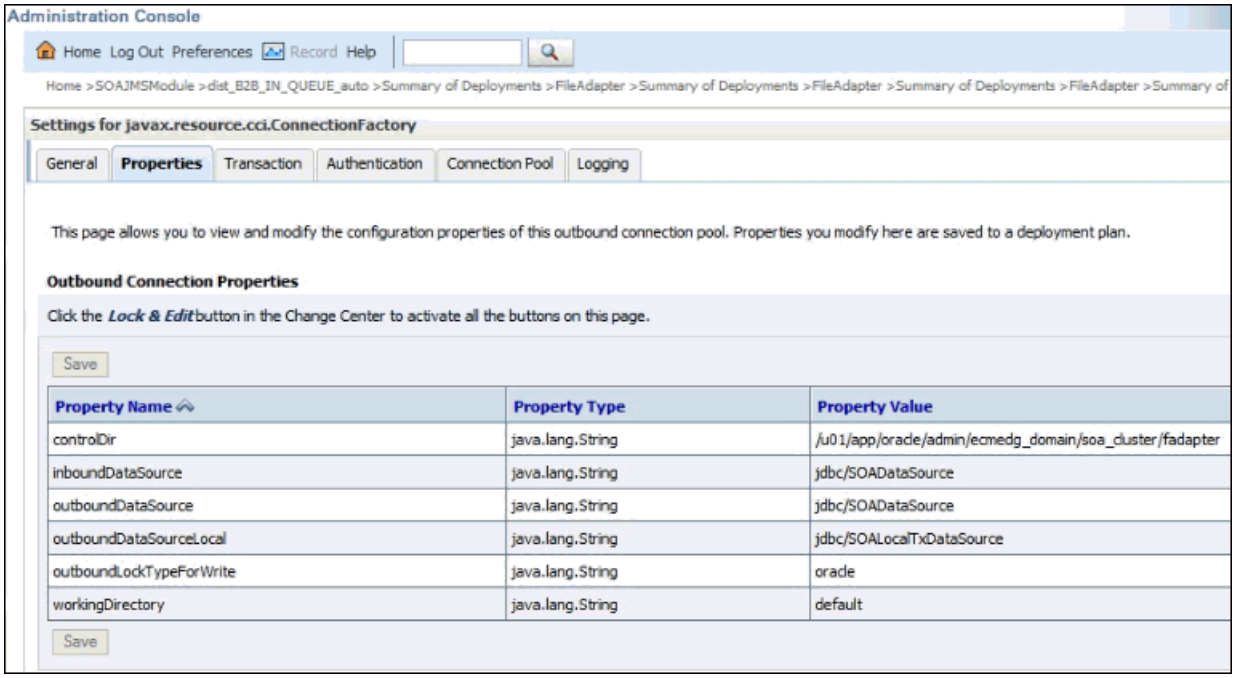
To make an outbound Oracle File or FTP Adapter service highly available using database table as a coordinator, you must modify the Oracle File Adapter deployment descriptor for the connection-instance corresponding to `eis/HAFileAdapter` in the WebLogic Server Administration Console.

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

To use the Database Mutex locking operation:

1. Log in to the WebLogic Server Administration Console. To access the console, navigate to `http://server_name:port_number/console`.
2. Click **Deployments** in the **Domain Structure** tree on the left.
3. Click **FileAdapter** under Summary of Deployments on the right pane.
4. Open the **Configuration** tab.
5. Open the **Outbound Connection Pools** tab, and expand **javax.resource.cci.ConnectionFactory** to see the configured connection factories.
6. Click **eis/HAFileAdapter**. The Outbound Connection Properties screen for the connection factory corresponding to high availability opens. The connection factory properties are displayed as shown in Figure 9–8.

Figure 9–8 WebLogic Server Administration Console - Settings for javax.resource.cci.ConnectionFactory Page



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

To reflect the updated values, click on the corresponding row for the **PropertyValue** column, and then press the Enter key on your keyboard. If you do not press the Enter key, the values will not be saved.

The new parameters in the connection factory for Oracle File and FTP Adapters follow:

- **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/SOADataSource`. 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/SOADataSource`. 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.

Note: The parameters available for FTP Adapters are slightly different than for the connection factory, but from a high-availability standpoint, just setting the control directory to a shared storage location is what matters.

---

8. Click **Save** after you update the properties.
9. On the Save Deployment Plan page, 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

10. Click **Save and Activate**.
11. Once the new deployment plan has been saved and activated, activate the FileAdapter deployment (the deployment remains in the **Prepared** state if not started). To activate the FileAdapter deployment plan:

    In the Administration Console, click **Deployments** in the **Domain Structure** tree on the left.

    Select the FileAdapter under **Summary of Deployments** on the right pane and Select **Start**, and then **Servicing All Requests**.

12. 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/HAFileAdapter" 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>
```

Notes:

- The location attribute is set to `eis/HAFileAdapter` for the connection factory.
- The preceding steps are for configuring the File Adapter for high availability. To configure the FTP Adapter for high availability, perform the same steps for updating the control directory. Use the `eis/Ftp/HAFtpAdapter` connection factory instance for these modifications.

9.9.2 Configuring the Oracle Database Adapter

Note: This step is optional and applies only to those deployments that require adapter support for the BPEL processes.

If you are using Logical Delete polling and you set `MarkReservedValue`, skip locking is not used. Formerly, the best practice for multiple Oracle Database Adapter process instances deployed to multiple Oracle BPEL Process Manager or Oracle Mediator nodes was essentially using `LogicalDeletePollingStrategy` or `DeletePollingStrategy` with a unique `MarkReservedValue` on each polling node, and setting `MaxTransactionSize`. However, with the introduction of skip locking in Oracle Fusion Middleware 11*g* Release 1 (11.1.2.0), that approach has now been superseded. If you were using this approach previously, you can simply remove (in db.jca) or clear (on the Logical Delete wizard page) the `MarkReservedValue` attribute, and you will automatically get skip locking.

The benefits of using skip locking over a reserved value include:

- Skip locking scales better in a cluster and under load.
- All work is in one transaction (as opposed to update/reserve, then commit, then select in a new transaction), so the risk of facing non-recoverable situations in high-availability environments is minimized.
- No unique `MarkReservedValue` needs to be specified. Previously, for this to work, you had to configure a complex variable, such as `R${weblogic.Name-2}-${IP-2}-${instance}`.

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

9.10 Configuring Node Manager for the WLS_SOA Managed Servers

Oracle recommends using host name verification for the communication between Node Manager and the servers in the domain. This requires the use of certificates for the different addresses communicating with the Administration Server and other servers. See Chapter 13, "Setting Up Node Manager" for further details. The procedures in that chapter must be performed twice using the following information:

Run	Host Name (*HOST*)	Virtual IP (*VIP*)	Server Name (*WLS_SERVER*)
Run 1:	SOAHOST1	SOAHOST1VHN1	WLS_SOA1
Run 2:	SOAHOST2	SOAHOST2VHN1	WLS_SOA2

9.11 Configuring Server Migration for the WLS_SOA Managed Servers

Server migration is required for proper failover of the Oracle SOA Suite components in the event of failure in any of the SOAHOST1 and SOAHOST2 nodes. See Chapter 14, "Configuring Server Migration for an Enterprise Deployment" for further details. For Oracle SOA Suite, use the following values for the variables in that chapter:

- Server names:
  - *WLS_SERVER1*: WLS_SOA1
  - *WLS_SERVER2*: WLS_SOA2
- Host names:
  - *HOST1*: SOAHOST1
  - *HOST2*: SOAHOST2
- Cluster name:
  - *CLUSTER*: SOA_Cluster

9.12 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 that 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 that guide. For information on how to recover components, see the "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" section in the guide. For information about database backup, see the *Oracle Database Backup and Recovery User's Guide*.

To back up the installation at this point:

1. Back up the web tier on WEBHOST1:
   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 Oracle instance on the web tier using the following command:

      ```
      tar -cvpf BACKUP_LOCATION/web_instance_name.tar ORACLE_INSTANCE
      ```

   d. Start the instance using `opmnctl`:

      ```
      cd 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 operating system tools such as tar for cold backups if possible.
3. Back up the Administration Server and Managed Server domain directories to save your domain configuration. The configuration files all exist in the *ORACLE_BASE*`/admin/`*domain_name* directory. Run the following command on SOAHOST1 to create the backup:

   ```
   tar -cvpf edgdomainback.tar ORACLE_BASE/admin/domain_name
   ```

10

Extending the Domain to Include Oracle WebCenter Content

This chapter describes how to extend a domain to include Oracle WebCenter Content using the Oracle Fusion Middleware Configuration Wizard. It contains the following sections:

Note: Before starting the setup process, read the *Oracle Fusion Middleware Release Notes* for your platform for additional installation and deployment information.

10.1 Overview of Extending the Domain to Include Oracle WebCenter Content

The Oracle WebCenter Content system is installed using the *WL_HOME* and ORACLE_HOME locations created in Chapter 6, "Installing the Software for an Enterprise Deployment," on a shared storage. WCCHOST1 and WCCHOST2 mount MW_HOME and reuse the existing Oracle WebLogic Server, Oracle SOA Suite, and Oracle WebCenter Content binary installations. The `pack` and `unpack` utilities are used to bootstrap the domain configuration for the WLS_WCC1 and WLS_WCC2 servers in these two new nodes. As a result, you do not need to install any software in these two nodes.

For the Oracle WebCenter Content system to work properly, WCCHOST1 and WCCHOST2 must maintain the same system requirements and configuration that was required for installing Oracle Fusion Middleware in SOAHOST1 and SOAHOST2. Otherwise, unpredictable behavior in the execution of binaries may occur.

Note: You will have already added Oracle SOA Suite components to the domain, as described in Section 9, "Extending the Domain to Include Oracle SOA Suite Components."

Extend the domain to include Oracle WebCenter Content. Table 10–1 lists the steps for configuring WebCenter Content and other tasks required for extending the domain with WebCenter Content Managed Servers.

Table 10–1 Steps for Extending the Domain with WebCenter Content

Step	Description	More Information
Extend the domain for WebCenter Content	Extend the WebLogic Server domain you created in Chapter 8, "Creating a Domain for an Enterprise Deployment," to include Oracle WebCenter Content.	Section 10.2, "Extending the Domain for WebCenter Content"
Start Node Manager on the WebCenter Content Managed Servers	Run the setNMProps.sh script and then start Node Manager on WCCHOST1 and on WCCHOST2.	Section 10.3, "Starting Node Manager on WCCHOST1 and WCCHOST2"
Propagate the domain configuration to the WebCenter Content Managed Servers	Propagate the start scripts and classpath configuration from the Administration Server's domain directory to the Managed Server domain directories.	Section 10.4, "Propagating the Domain Configuration to the Managed Server Domain Directories"
Restart the Administration Server for the domain	Stop and then restart the Administration Server.	Section 10.5, "Restarting the Administration Server"
Start the first WebCenter Content Managed Server and configure its Content Server instance	Start the WLS_WCC1 Managed Server and complete the initial Content Server configuration.	Section 10.6, "Starting the WLS_WCC1 Managed Server and Configuring Content Server"
Propagate the changes in the cwallet.sso file back to the Administration Server.	Copy the updated cwallet.sso file to the Administration Server directory.	Section 10.7, "Updating the cwallet File in the Administration Server"
Start the second WebCenter Content Managed Server and configure its Content Server instance	Configure the WLS_WCC2 Managed Server and complete the initial Content Server configuration.	Section 10.8, "Starting the WLS_WCC2 Managed Server and Configuring Content Server"
Verify the configuration of GridLink data sources and Oracle Notification Service (ONS)	Follow these instructions to verify that the configuration of GridLink data sources and ONS is correct.	Section 10.9, "Validating GridLink Data Sources for WebCenter Content"

Table 10–1 (Cont.) Steps for Extending the Domain with WebCenter Content

Step	Description	More Information
Configure parameters for cluster nodes	Add directory locations for WebCenter Content trace and event logs, as needed.	Section 10.10, "Configuring Additional Parameters"
Enable service retries after an Oracle RAC failover	Set the ServiceAllowRetry configuration parameter to `true` in the Content Server `config.cfg` file.	Section 10.11, "Configuring Service Retries for Oracle WebCenter Content"
Configure Oracle HTTP Server with the extended domain	Configure the Oracle HTTP Server with the Managed Servers, set the front-end HTTP host and port, and set the WebLogic Server cluster address for WCC_Cluster.	Section 10.12, "Configuring Oracle HTTP Server for the WLS_WCC Managed Servers"
Validate access to WebCenter Content through Oracle HTTP Server	Verify the URLs to ensure that appropriate routing and failover is working from Oracle HTTP Server to WCC_Cluster.	Section 10.13, "Validating Access Through the Load Balancer"
Configure host name verification for the communication between Node Manager and the Managed Servers in the domain	Use certificates for the different addresses communicating with the Administration Server and other servers.	Section 10.14, "Configuring Node Manager for the WLS_WCC and WLS_IMG Managed Servers"
Back Up the WebCenter Content Configuration	Back up the newly extended domain configuration.	Section 10.15, "Backing Up the Installation"

10.2 Extending the Domain for WebCenter Content

You can extend the domain created in Section 8, "Creating a Domain for an Enterprise Deployment" to include Oracle WebCenter Content. The instructions in this section assume that the Oracle WebCenter Content deployment uses the same database service as the Oracle SOA Suite deployment (`wccedg.mycompany.com`).

> **Note:** Before performing these steps, back up the domain as described in the *Oracle Fusion Middleware Administrator's Guide*.

To extend the domain for Oracle WebCenter Content:

1. Make sure that the database where you installed the repository is running.

   For Oracle RAC databases, it is recommended that all instances are running, so that the validation check later on becomes more reliable.

2. On SOAHOST1, change the directory to the location of the Oracle Fusion Middleware Configuration Wizard. This is within the Oracle Common home directory (domain extensions are run from the node where the Administration Server resides).

   ```
   cd ORACLE_COMMON_HOME/common/bin
   ```

3. Start the Fusion Middleware Configuration Wizard:

   ```
   ./config.sh
   ```

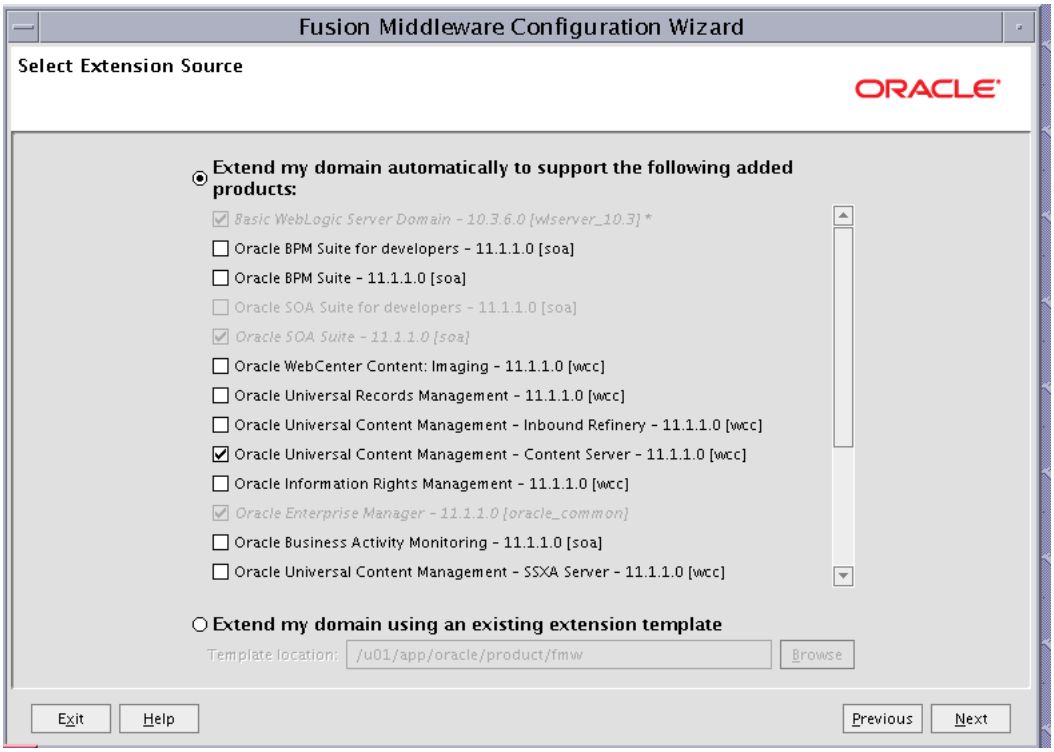
4. In the Welcome screen, select **Extend an existing WebLogic domain**, and click **Next**.

5. In the Select a WebLogic Domain Directory screen, select the WebLogic Server domain directory (*ORACLE_BASE*/admin/*domain_name*/aserver/*domain_name*), and click **Next**.
6. In the Select Extension Source screen (Figure 10–1), make these selections:
   - Select **Extend my domain automatically to support the following added products**.
   - Select this product:

     Oracle Universal Content Management - Content Server - 11.1.1.0 [wcc]

     This is the selection for Oracle WebCenter Content.

Figure 10–1 Select Extension Source Screen for Oracle WebCenter Content



The following products should already be selected and grayed out. They were selected when you created the domain (Section 8.3) or extended it for Oracle SOA Suite components (Section 9.3).

Basic WebLogic Server Domain - 10.3.6.0 [wlserver_10.3]

Oracle SOA Suite for developers - 11.1.1.0 [soa]

Oracle SOA Suite - 11.1.1.0 [soa]

Oracle Enterprise Manager - 11.1.1.0 [oracle_common]

Oracle WSM Policy Manager - 11.1.1.0 [oracle_common]

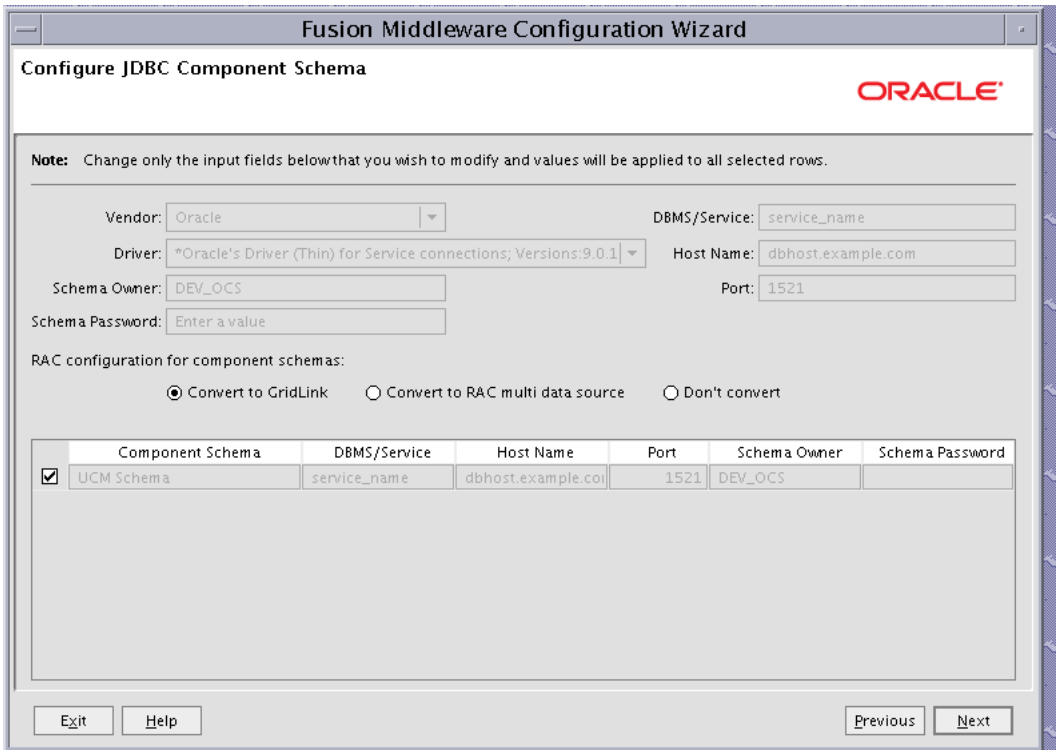
Oracle JRF - 11.1.1.0 [oracle_common]

Click **Next**.

7. In the Configure JDBC Component Schema screen (Figure 10–2), do the following:

   a. Select **UCM Schema** (for WebCenter Content). Do not select any of the other existing schemas.

Figure 10–2 Configure JDBC Component Schema Screen for Oracle WebCenter Content



   b. For the RAC configuration, you can select **Convert to GridLink** or **Convert to RAC multi data source** (described in Appendix A, "Using Multi Data Sources with Oracle RAC"). For the instructions given here, select **Convert to GridLink**.

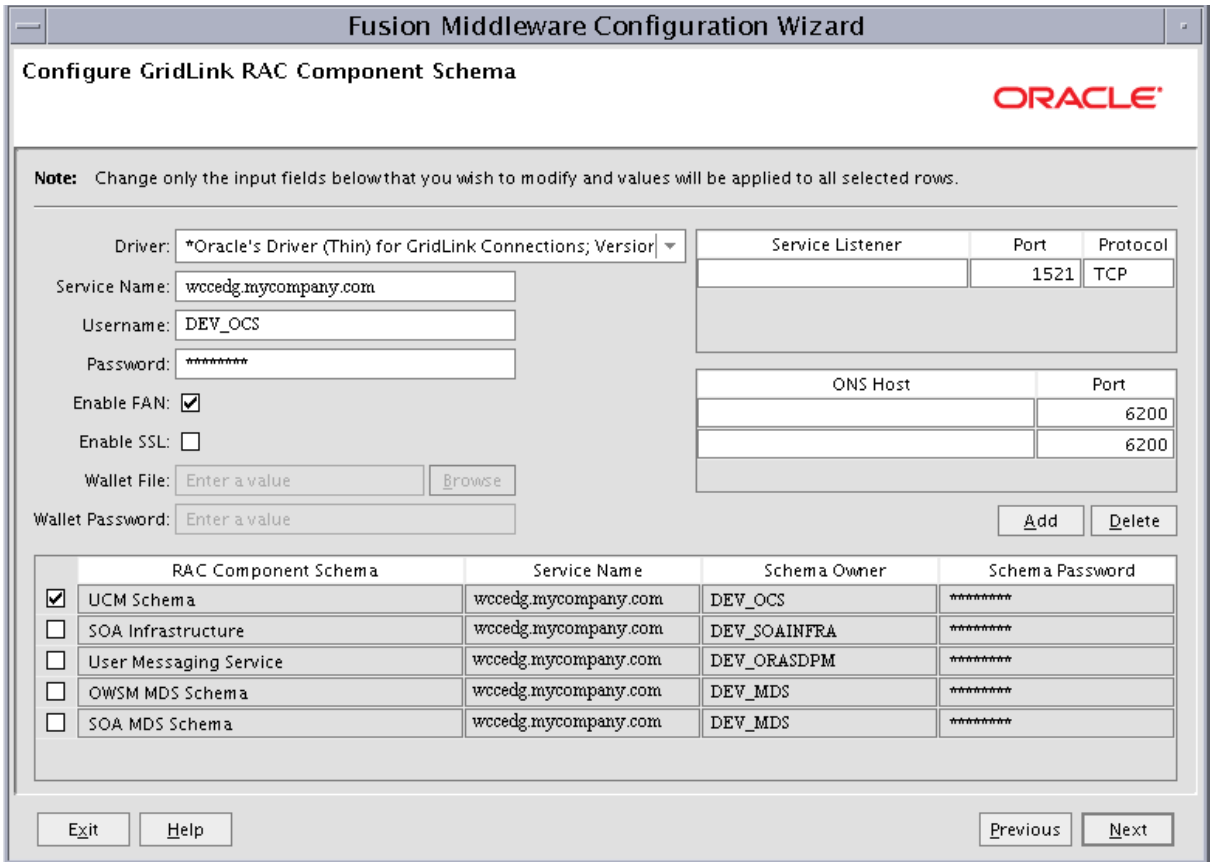
      After you select a RAC configuration, all selected schemas are grayed.

   c. Click **Next**.

8. In the Configure GridLink RAC Component Schema screen (Figure 10–3), do the following:

   a. Select **UCM Schema** (for WebCenter Content). Leave the other data sources as they are.

Figure 10–3 Configure GridLink RAC Component Schema Screen for WebCenter Content



b. Enter values for the following fields, specifying the connection information for the GridLink RAC database that was seeded through RCU:

- **Driver:** Select **Oracle driver (Thin) for GridLinkConnections; Versions:10 and later**.
- **Service Name:** Enter the service name of the Oracle RAC database in lowercase letters, followed by the domain name; for example, `wccedg.mycompany.com`.
- **Username:** Enter the complete user name for the database schema owner of the corresponding component.

  This book uses `DEV` as the prefix of user names for the database schemas.
- **Password:** Enter the password for the database schema owner.
- Select **Enable FAN**.
- **Enable SSL:** Leave this option deselected.

  If you select SSL to enable Oracle Notification Service (ONS) notification encryption, provide the appropriate **Wallet File** and **Wallet Password** details.

– **Service listener:** Enter the Oracle Single Client Access Name (SCAN) address and port for the Oracle RAC database being used. The protocol should be `TCP`.

  Oracle recommends that you use a SCAN address to specify the Service Listener (and OSN Host) so you do not need to update a GridLink data source containing a SCAN address if you add or remove Oracle RAC nodes. To determine the SCAN address, query the `remote_listener` parameter in the database:

```
SQL>show parameter remote_listener;

NAME              TYPE        VALUE
-----             ------      -------
remote_listener   string      db-scan.mycompany.com:1521
```

Note: For Oracle Database 11*g* Release 1 (11.1), use the virtual IP and port of each database instance listener; for example:

```
custdbhost1-vip.mycompany.com (port 1521)
```

and

```
custdbhost2-vip.mycompany.com (1521)
```

For Oracle Database 10*g*, use multi data sources to connect to an Oracle RAC database. For information about configuring multi data sources, see Appendix A, "Using Multi Data Sources with Oracle RAC."

– **ONS Host:** Enter here also the SCAN address for the RAC database and the ONS remote port, as reported by the database:

```
[orcl@CUSTDBHOST2~]$ srvctl config nodeapps -s
ONS exists: Local port 6100, remote port 6200, EM port 2016
```

Note: For Oracle Database 11*g* Release 1 (11.1), use the host name and port of each database's ONS service; for example:

```
custdbhost1.mycompany.com (port 6200)
```

and

```
custdbhost2.mycompany.com (6200)
```

---

c. Click **Next**.

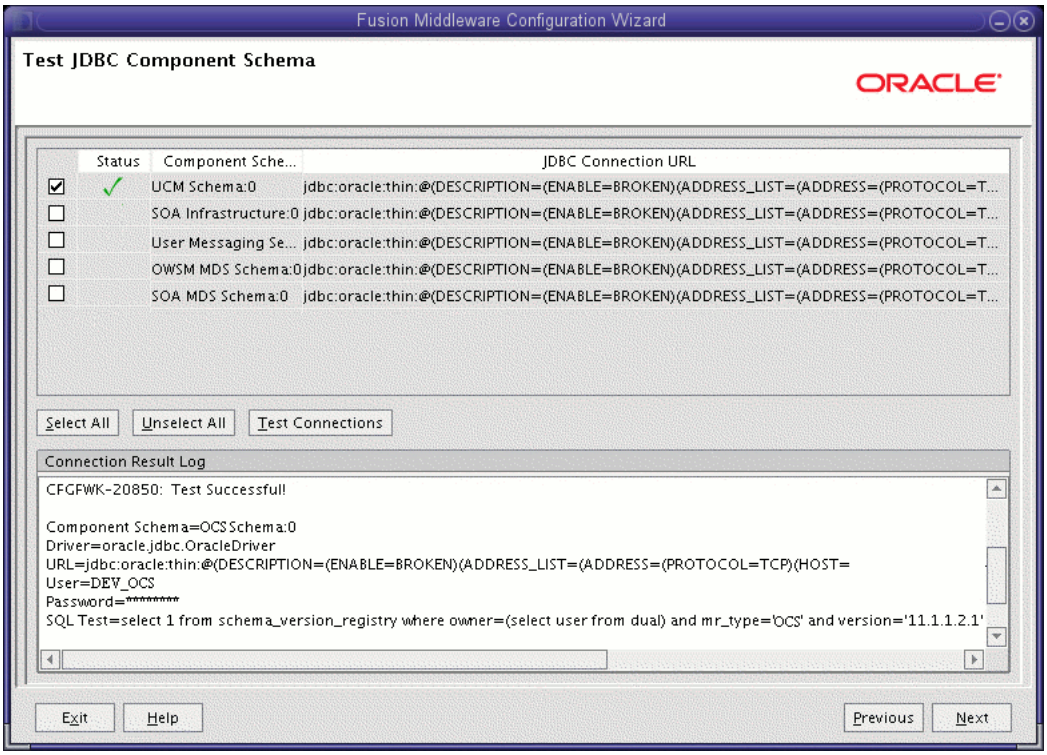
Note: Leave the **SOA Infrastructure**, **User Messaging Service**, **OWSM MDS Schema**, and **SOA MDS Schema** information as is.

---

9. In the Test JDBC Component Schema screen (Figure 10–4), select the **UCM Schema** row, then click **Test Connections**.

   The **Connection Results Log** displays the results. Ensure that the connection to the database that contains the schema was successful. If not, click **Previous** to return to the previous screen, correct your entry, and then retry the test.

Figure 10–4 Test JDBC Component Schema Screen for WebCenter Content



   Click **Next** when the connection is successful.

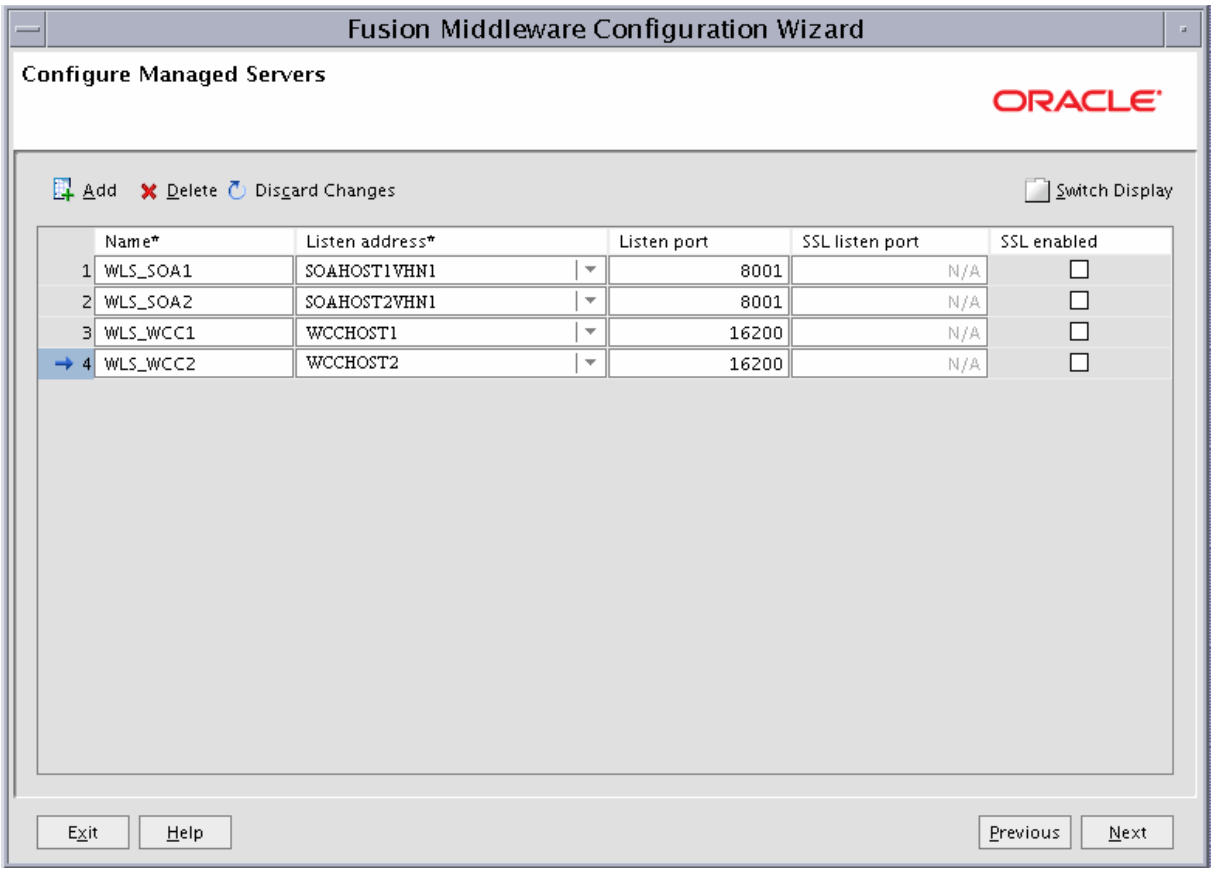
10. In the Optional Configuration screen, select the following options:

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

    Click **Next**.

11. In the Configure Managed Servers screen (Figure 10–5), add the required Managed Servers.

Figure 10–5 Configure Managed Servers for WebCenter Content



A server is created automatically. Rename this server to WLS_WCC1 and add a new server called WLS_WCC2. Give these servers the attributes listed in Table 10–2. Do not modify the other servers that are shown in this screen; leave them as they are.

Table 10–2 Managed Servers

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled
WLS_WCC1	WCCHOST1	16200	n/a	No
WLS_WCC2	WCCHOST2	16200	n/a	No

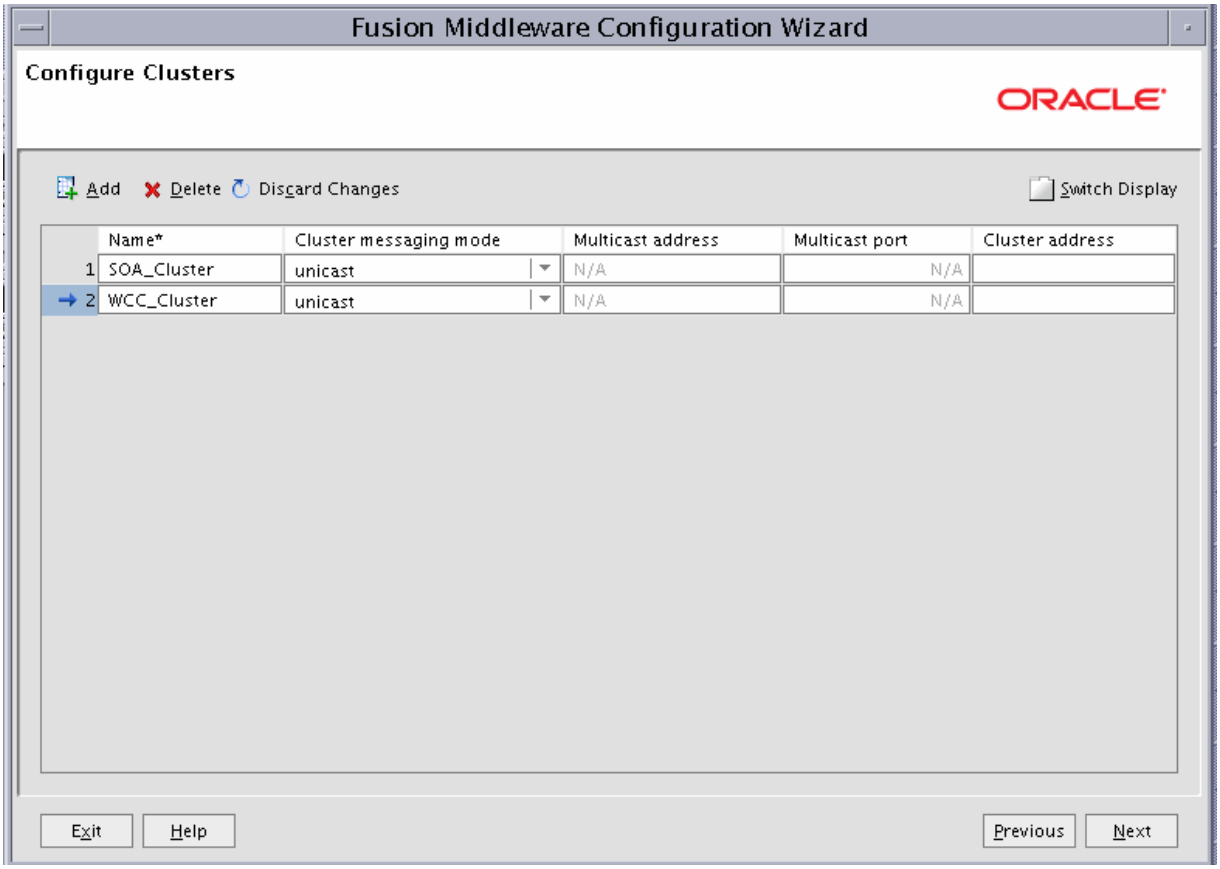
Click **Next**.

12. In the Configure Clusters screen (Figure 10–6, click **Add** to add the clusters as shown in Table 10–3. Do not modify the other clusters that appear in this screen; leave them as they are.

Table 10–3 Clusters

Name	Cluster Messaging Mode	Multicast Address	Multicast Port	Cluster Address
WCC_Cluster	unicast	n/a	n/a	Leave empty

Figure 10–6 Configure Clusters Screen for WebCenter Content



Click **Next**.

13. In the Assign Servers to Clusters screen, add the following. Do not modify the other assignments that appear in this screen; leave them as they are.
    - **WCC_Cluster:**
      - WLS_WCC1
      - WLS_WCC2

    Click **Next**.

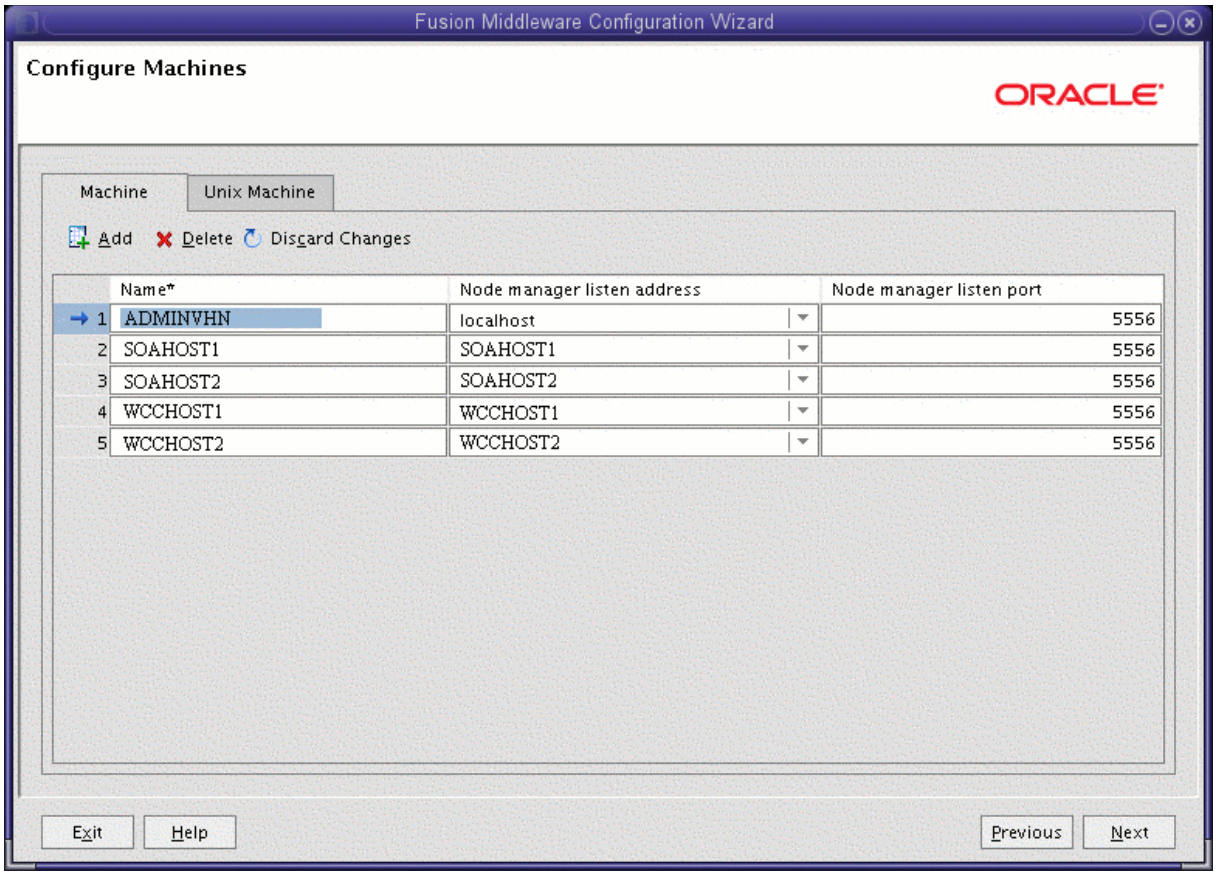
14. In the Configure Machines screen (Figure 10–7), open the **Unix Machine** tab and add the following two new machines:

Table 10–4 Machines

Name	Node Manager Listen Address
WCCHOST1	WCCHOST1
WCCHOST2	WCCHOST2

Leave all other fields set to their default values.

Figure 10–7 Configure Machines Screen

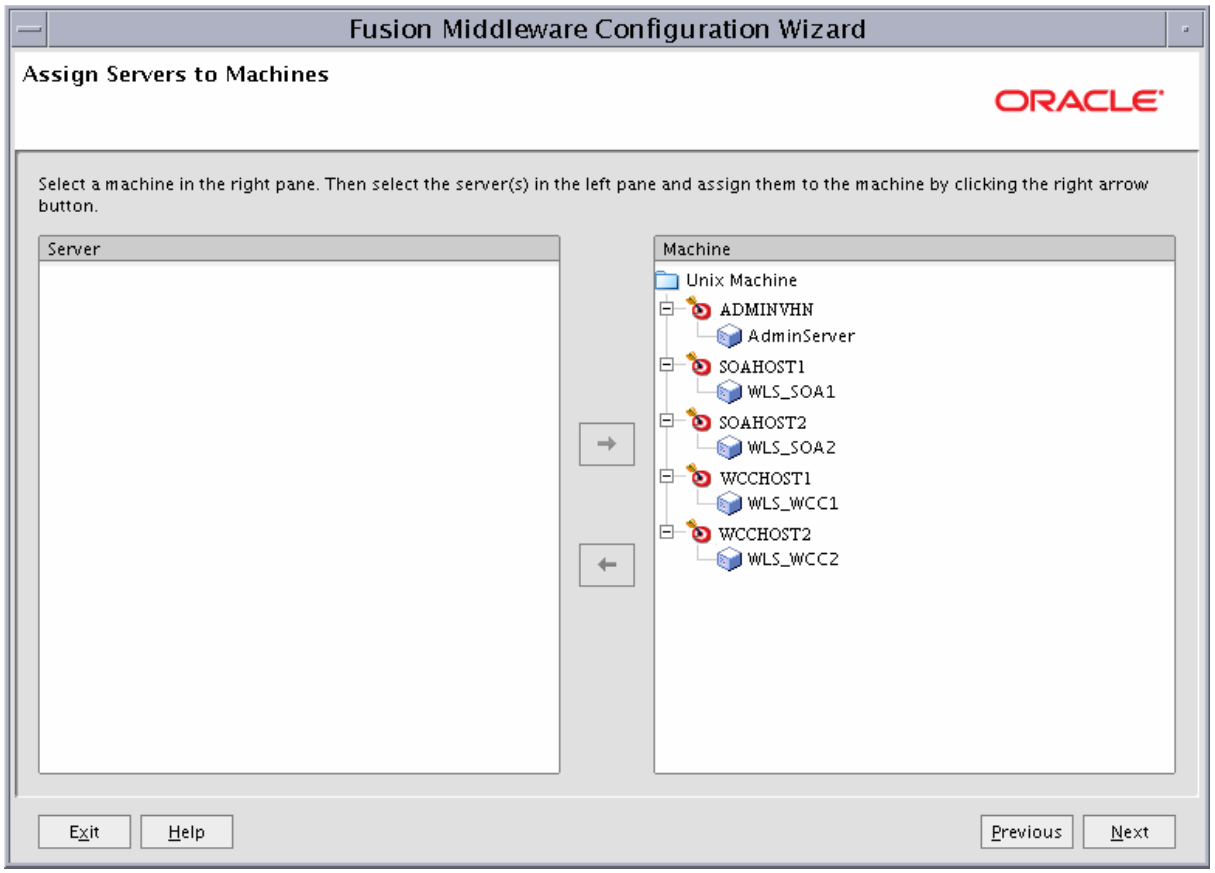


Click **Next**.

15. In the Assign Servers to Machines screen (Figure 10–8), assign servers to machines as follows:

    - Assign **WLS_WCC1** to **WCCHOST1**.
    - Assign **WLS_WCC2** to **WCCHOST2**.

Figure 10–8 Assign Server to Machines Screen for WebCenter Content



Click **Next**.

16. In the Target Deployments to Clusters or Servers screen, make sure that the DMS Application is targeted to the SOA_Cluster, WCC_Cluster, and Admin Server. Click **Next**.
17. In the Target Services to Clusters or Servers screen, click **Next**.
18. In the Configuration Summary screen, click **Extend**.
19. If you see a warning dialog about port conflicts for the domain, click **OK**.
20. In the Creating Domain screen, click **Done**.
21. Restart the Administration Server to make these changes to take effect.

    To restart the Administration Server, stop it first using the Administration Console and then start it again as described in Section 8.4.3, "Starting the Administration Server on SOAHOST1."

10.3 Starting Node Manager on WCCHOST1 and WCCHOST2

To start Node Manager on WCCHOST1 and WCCHOST2 if Node Manager has not started already:

1. On both WCCHOST1 and WCCHOST2, 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:

   ```
   cd ORACLE_COMMON_HOME/common/bin

   ./setNMProps.sh
   ```

   Note: You must use the `StartScriptEnabled` property to avoid class loading failures and other problems. See also Section 16.12.3, "Incomplete Policy Migration After Failed Restart of SOA Server."

   Note: If the WebCenter Content server is sharing the Middleware home in a local or shared storage with Oracle SOA Suite, as suggested in the configuration described in Chapter 3, "Preparing the Network for an Enterprise Deployment" it is not required to run setNMProps.sh again. In this case, Node Manager has already been configured to use a start script.

2. Run the following commands on both WCCHOST1 and WCCHOST2 to start Node Manager:

   ```
   cd WL_HOME/server/bin

   ./startNodeManager.sh
   ```

10.4 Propagating the Domain Configuration to the Managed Server Domain Directories

To propagate the domain configuration:

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

   ```
   cd ORACLE_COMMON_HOME/common/bin

   ./pack.sh -managed=true -domain=ORACLE_BASE/admin/domain_name/aserver/domain_
   name -template=edgdomaintemplateWCC.jar -template_name=edgdomain_templateWCC
   ```

2. Run the following command on SOAHOST1 to copy the template pack created in the previous step to WCCHOST2:

   Note: Assuming that WCCHOST1 shares the *ORACLE_HOME* with SOAHOST1, the template will be present in the same directory in WCCHOST1; otherwise, copy it also to WCCHOST1.

   ```
   scp edgdomaintemplateWCC.jar oracle@WCCHOST2:ORACLE_BASE/product/fmw/oracle_
   common/common/bin
   ```

3. Run the `unpack` command on WCCHOST1 to unpack the propagated template.

> **Note:** Make sure to run `unpack` from the *ORACLE_COMMON_HOME*`/common/bin/` directory, not from *WL_HOME*`/common/bin/`.

```
cd ORACLE_COMMON_HOME/common/bin

./unpack.sh -domain=ORACLE_BASE/admin/domain_name/mserver/domain_name
-template=edgdomaintemplateWCC.jar -app_dir=ORACLE_BASE/admin/domain_
name/mserver/applications
```

> **Note:** The *ORACLE_BASE*`/admin/`*domain_name*`/mserver/` directory must exist before you run `unpack`. In addition, the *ORACLE_BASE*`/admin/`*domain_name*`/mserver/applications/` directory must be empty.

4. Repeat step 3 for WCCHOST2.

10.5 Restarting the Administration Server

Restart the Administration Server to make these changes take effect. To restart the Administration Server, stop it first using the Administration Console and then start it again as described in Section 8.4.3, "Starting the Administration Server on SOAHOST1."

10.6 Starting the WLS_WCC1 Managed Server and Configuring Content Server

After you start the WLS_WCC1 Managed Server, you can configure Oracle WebCenter Content Server through the web interface.

To start the WLS_WCC1 Managed Server:

1. Start the WLS_WCC1 Managed Server using the Oracle WebLogic Server Administration Console as follows:
   a. Expand the Environment node in the **Domain Structure** tree on the left.
   b. Click **Servers**.
   c. On the Summary or Servers page, open the **Control** tab.
   d. Select **WLS_WCC1** and then click **Start**.
2. Verify that the server status is reported as `Running` in the Administration 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. For possible causes, see Section 16.12, "Troubleshooting the Oracle WebCenter Content Enterprise Deployment Topology."

To configure Content Server:

1. Log in to WLS_WCC1 at `http://WCCHOST1:16200/cs` using your WebLogic Server administration user name and password to display a configuration page.

   Note: The WebCenter Content configuration files are on a shared disk so that all members of the cluster can access them. The shared disk location of the Oracle WebCenter Content enterprise deployment is at *ORACLE_BASE*`/admin/`*domain_name*`/`*wcc_cluster_name*`/`.

2. Change the following values on the server configuration page (make sure to select the **Is New Content Server Instance** checkbox to see all options):

   - **Content Server Instance Folder:** Set this to *ORACLE_BASE*`/admin/`*domain_name*`/`*wcc_cluster_name*`/cs`.
   - **Native File Repository Location:** Set this to *ORACLE_BASE*`/admin/`*domain_name*`/`*wcc_cluster_name*`/cs/vault`.
   - **WebLayout Folder:** Set this to *ORACLE_BASE*`/admin/`*domain_name*`/`*wcc_cluster_name*`/cs/weblayout`.
   - **User Profile Folder:** Set this to *ORACLE_BASE*`/admin/`*domain_name*`/`*wcc_cluster_name*`/cs/data/users/profiles`.
   - **Server Socket Port:** Set this to `4444`.
   - **Incoming Socket Connection Address Security Filter:** Set this to a pipe-delimited list of the local host and the server IPs:

     ```
     127.0.0.1|WCCHOST1|WCCHOST2|WEBHOST1|WEBHOST2
     ```

     Note: This will be changed later to a host name-based list (see Section 11.16, "Adding the Imaging Server Listen Addresses to the List of Allowed Hosts in Oracle WebCenter Content"). At this point, we need the connections to be allowed for operations that will be done before the security filter is changed to host names.

   - **WebServer HTTP/HTTPS Address:** Set this to `wcc.mycompany.com:443`.
   - **Web Address is HTTPS:** Select this checkbox.
   - **Server Instance Name:** Set this to `WCC_Cluster1`.
   - **Server Instance Label:** Set this to `WCC_Cluster1`.
   - **Server Instance Description:** Set this to `Cluster wcc_cluster1`.
   - **Auto_Number Prefix:** Set this to `wcc_cluster1-`.

3. Click **Submit** when finished, and restart the Managed Server using the WebLogic Server Administration Console.

10.7 Updating the cwallet File in the Administration Server

Oracle WebCenter Content Server updates the `cwallet.sso` file located in *ORACLE_BASE*`/admin/`*domain_name*`/mserver/`*domain_name*`/config/fmwconfig/` when it starts. This change needs to be propagated back to the Administration Server. To do this, copy the file to *ORACLE_BASE*`/admin/`*domain_name*`/aserver/`*domain_name*`/config/fmwconfig` on SOAHOST1 using the following command on WCCHOST1 (all on a single line):

```
scp ORACLE_BASE/admin/domain_name/mserver/domain_name/config/fmwconfig/cwallet.sso
oracle@SOAHOST1:ORACLE_BASE/admin/domain_name/aserver/domain_
name/config/fmwconfig/
```

Note: If any operation is performed in a WLS_WCC*n* server that modifies the `cwallet.sso` file in the *ORACLE_BASE*`/admin/`*domain_name*`/mserver/`*domain_name*`/config/fmwconfig/` directory, the file will have to be immediately copied to the Administration Server domain directory on SOAHOST1 at *ORACLE_BASE*`/admin/`*domain_name*`/aserver/`*domain_name*`/config/fmwconfig/`.

10.8 Starting the WLS_WCC2 Managed Server and Configuring Content Server

After you start the WLS_WCC2 Managed Server, you can configure Oracle WebCenter Content Server through the web interface.

To start the WLS_WCC2 Managed Server:

1. Start the WLS_WCC2 Managed Server using the WebLogic Server Administration Console, as follows:
   a. Expand the Environment node in the **Domain Structure** tree on the left.
   b. Click **Servers**.
   c. On the Summary of Servers page, open the **Control** tab.
   d. Select **WLS_WCC2**, and then click **Start**.
2. Verify that the server status is reported as `Running` in the Administration 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. For possible causes, see Section 16.12, "Troubleshooting the Oracle WebCenter Content Enterprise Deployment Topology."

To configure Content Server:

1. Log in to WLS_WCC2 at `http://WCCHOST2:16200/cs` using your Oracle WebLogic administration user name and password to display a configuration page.

   Note: The Oracle WebCenter Content configuration files are on a shared disk so that all members of the cluster can access them. The shared disk location of the Oracle WebCenter Content enterprise deployment is at *ORACLE_BASE*/admin/*domain_name*/*wcc_cluster_name*/.

2. Change the following values on the server configuration page:
   - **Content Server Instance Folder:** Set this to *ORACLE_BASE*/admin/*domain_name*/*wcc_cluster_name*/cs.
   - **Native File Repository Location:** Set this to *ORACLE_BASE*/admin/*domain_name*/*wcc_cluster_name*/cs/vault.
   - **WebLayout Folder:** Set this to *ORACLE_BASE*/admin/*domain_name*/*wcc_cluster_name*/cs/weblayout.
   - **User Profile Folder:** Set this to *ORACLE_BASE*/admin/*domain_name*/*wcc_cluster_name*/cs/data/users/profiles.
   - **Content Server URL Prefix:** /cs/ (default value)

   Make sure that the **Is new Content Server Instance?** checkbox is not selected.

3. Click **Submit** when finished, and restart the Managed Server using the WebLogic Server Administration Console.

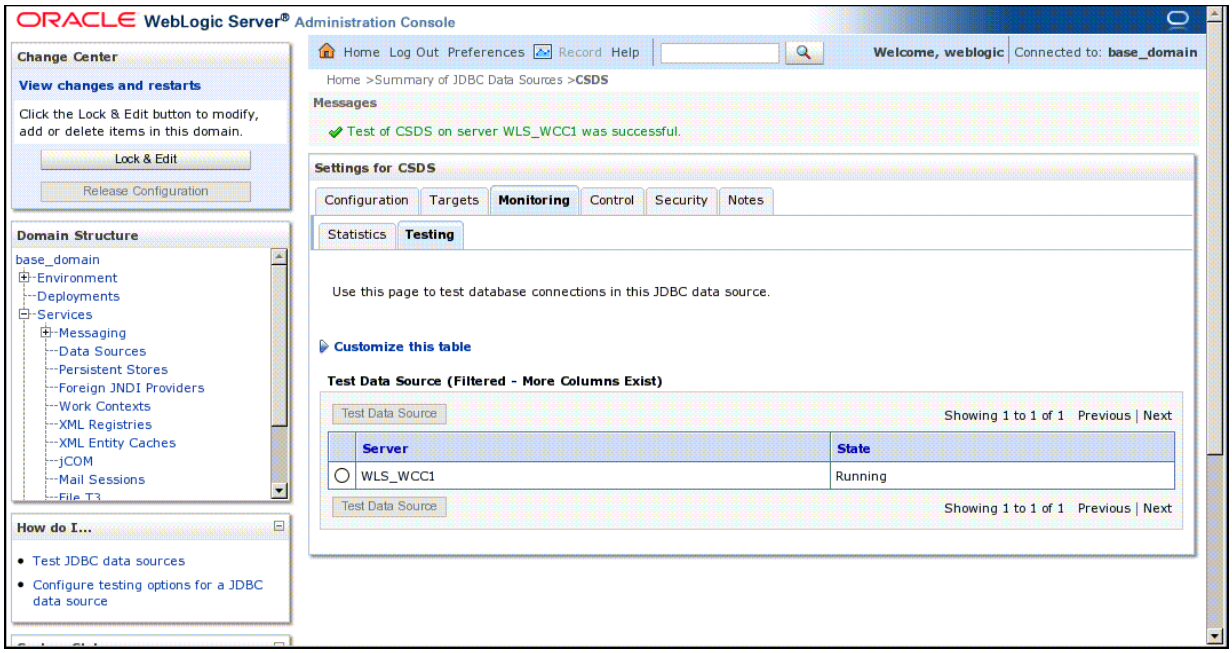
10.9 Validating GridLink Data Sources for WebCenter Content

After the servers are started, verify that the GridLink data sources are correctly configured and that the ONS setup is correct. Perform these procedures for every GridLink data source created.

To verify the configuration of a GridLink data source for WebCenter Content:

1. Log in to the WebLogic Server Administration Console.
2. In the **Domain Structure** tree, expand **Services**, then click **Data Sources**.
3. Click the name of a GridLink data source that was created.
4. Click the **Monitoring** tab.
5. Click the **Testing** tab (Figure 10–9), select one of the servers, and click **Test Data Source**.

Figure 10–9 Testing a GridLink Data Source for WebCenter Content



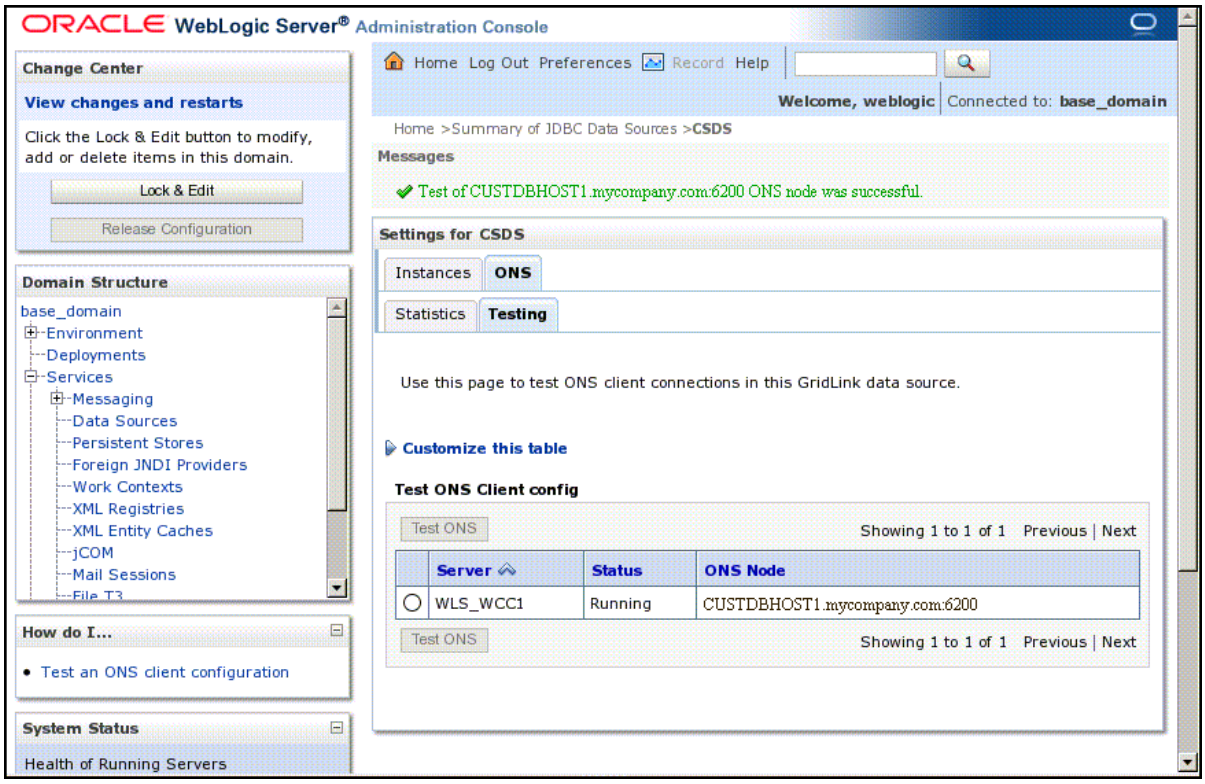
The test should be successful if the configuration is correct.

6. Repeat the test for every WebLogic Server instance that uses the GridLink data source.

To verify the configuration of ONS for a GridLink data source for WebCenter Content:

1. Log in to the WebLogic Server Administration Console.
2. In the **Domain Structure** tree, expand **Services**, then click **Data Sources**.
3. Click the name of a GridLink data source.
4. Click the **Monitoring** tab.
5. Click the **ONS** tab and then the **Testing** tab (Figure 10–10).
6. Select a server, and click **Test ONS**.

Figure 10–10 Testing the ONS Configuration for WebCenter Content



The test should be successful if the configuration is correct. If the ONS test fails, verify that the ONS service is running in the Oracle RAC database nodes:

```
[orcl@CUSTDBHOST1 ~]$ srvctl status scan_listener
SCAN Listener LISTENER_SCAN1 is enabled
SCAN listener LISTENER_SCAN1 is running on node CUSTDBHOST1
SCAN Listener LISTENER_SCAN2 is enabled
SCAN listener LISTENER_SCAN2 is running on node CUSTDBHOST2
SCAN Listener LISTENER_SCAN3 is enabled
SCAN listener LISTENER_SCAN3 is running on node CUSTDBHOST2


[orcl@CUSTDBHOST1 ~]$ srvctl config nodeapps -s
ONS exists: Local port 6100, remote port 6200, EM port 2016


[orcl@CUSTDBHOST1 ~]$ srvctl status nodeapps | grep ONS
ONS is enabled
ONS daemon is running on node: CUSTDBHOST1
ONS daemon is running on node: CUSTDBHOST2
```

7. Repeat the ONS test for every WebLogic Server instance that uses the GridLink data source.

10.10 Configuring Additional Parameters

Using a text editor, add the following options to each cluster node's *MSERVER_HOME*/base_domain/ucm/cs/bin/intradoc.cfg file, where the directories specified are on a direct-bus-attached-controlled *local* disk and not a remote file system, such as a UNIX/Linux mounted NFS or clustered file system (like OCFS2, GFS2, or GPFS):

```
TraceDirectory=ORACLE_BASE/admin/domain_name/mserver/domain_name/servers/WLS_WCCN/logs
EventDirectory=ORACLE_BASE/admin/domain_name/mserver/domain_name/servers/WLS_WCCN/logs/event/
ArchiverDoLocks=true
DisableSharedCacheChecking=true
```

The trailing *N* should match your nodes' server names, like WLS_WCC1 is WCCHOST1 and WLS_WCC2 is WCCHOST2, and so on.

These changes will take effect after a restart of all WebCenter Content Managed Servers, at the end of the procedure described in Section 10.11, "Configuring Service Retries for Oracle WebCenter Content."

Note: The directories can reside in any local disk path that you have determined to have enough space to hold the WebCenter Content logs and any trace that you may configure. The preceding paths are a suggestion.

10.11 Configuring Service Retries for Oracle WebCenter Content

The following parameter should be set in the Content Server config.cfg file to enable login retries during an Oracle RAC failover:

```
ServiceAllowRetry=true
```

If this value is not set, users will need to manually retry any operation that was in progress when the failover began.

To add the configuration parameter for Oracle WebCenter Content:

1. Go to Oracle WebCenter Content Server at http://WCCHOST1:16200/cs, and log in using your WebLogic Server administration user name and password.
2. Open the Administration page, and then choose **Admin Server**.
3. On the Content Admin Server page, click **General Configuration** on the left.
4. On the General Configuration page, add the following parameter in the Additional Configuration Variables box:

   ```
   ServiceAllowRetry=true
   ```

5. Click **Save**, and restart all WebCenter Content Managed Servers.

Note: The new parameter is included in the config.cfg file, which is at the following location:

ORACLE_BASE/admin/*domain_name*/*wcc_cluster_name*/cs/config/config.cfg

(You can also edit this file directly in a text editor. Do not forget to restart all WebCenter Content Managed Servers.)

10.12 Configuring Oracle HTTP Server for the WLS_WCC Managed Servers

To enable Oracle HTTP Server to route to WCC_Cluster, which contain the WLS_WCC1 and WLS_WCC2 Managed Servers, you must set the `WebLogicCluster` parameter to the list of nodes in the cluster:

1. For each of the web servers on WEBHOST1 and WEBHOST2, add the following lines to the *ORACLE_INSTANCE*`/config/OHS/ohs1/moduleconf/wcc_vh.conf` and *ORACLE_INSTANCE*`/config/OHS/ohs2/moduleconf/wcc_vh.conf` files:

```
# UCM
<Location /cs>
   WebLogicCluster WCCHOST1:16200,WCCHOST2:16200
   SetHandler weblogic-handler
   WLCookieName JSESSIONID
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>

<Location /adfAuthentication>
   WebLogicCluster WCCHOST1:16200,WCCHOST2:16200
   SetHandler weblogic-handler
   WLCookieName JSESSIONID
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>

<Location /_ocsh>
   WebLogicCluster WCCHOST1:16200,WCCHOST2:16200
   SetHandler weblogic-handler
   WLCookieName JSESSIONID
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>
```

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

   ORACLE_BASE/admin/*instance_name*/bin/opmnctl restartproc ias-component=ohsX

   For WEBHOST1, use `ohs1` for `ias-component` and for WEBHOST2 use `ohs2`.

10.13 Validating Access Through the Load Balancer

You should verify URLs to ensure that appropriate routing and failover is working from Oracle HTTP Server to WCC_Cluster. To verify the URLs:

1. While WLS_WCC2 is running, stop WLS_WCC1 using the WebLogic Server Administration Console.
2. Access `http://wcc.mycompany.com/cs` to verify it is functioning properly.
3. Start WLS_WCC1 from the WebLogic Server Administration Console.
4. Stop WLS_WCC2 from the WebLogic Server Administration Console.
5. Access `http://wcc.mycompany.com/cs` to verify it is functioning properly.

10.14 Configuring Node Manager for the WLS_WCC and WLS_IMG Managed Servers

Oracle recommends using host name verification for the communication between Node Manager and the servers in the domain. This requires the use of certificates for the different addresses communicating with the Administration Server and other servers. For more details, see Chapter 13, "Setting Up Node Manager." The procedures in that chapter must be performed twice using the information in the following table.

Run	Host Name (*HOST*)	Virtual IP (*VIP*)	Server Name (*WLS_SERVER*)
Run 1	WCCHOST1	WCCHOST1VHN1*	WLS_WCC1
Run 2	WCCHOST2	WCCHOST2VHN1*	WLS_WCC2

* Optional; required only for Oracle WebCenter Content: Imaging Managed Servers (WLS_IMG).

Notes:

- Even though the WLS_IMG Managed Servers are not yet configured at this point and are not mandatory for an Oracle WebCenter Content configuration that includes only WebCenter Content, the virtual host names used by Oracle WebCenter Content: Imaging are configured here to provide a one-step configuration process that includes both types of servers. For information about configuring the WLS_IMG Managed Server, see Chapter 11, "Extending the Domain to Include Imaging."
- For server migration of WLS_IMG1 and WLS_IMG2, you need to associate these servers with virtual host names (WCCHOST1VHN1 and WCCHOST2VHN1). Check that these virtual host names are enabled by DNS or `/etc/hosts` resolution in your system and that they map to the appropriate VIPs. For more information, see Section 3.4, "Configuring IPs and Virtual IPs."
- Perform all steps in Chapter 13, "Setting Up Node Manager," except for Section 13.3.5, "Configuring Managed Servers to Use the Custom Keystores," for the WLS_IMG servers. This step can be done after the WLS_IMG servers are added to the domain.

10.15 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 that 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 that guide. For information on how to recover components, see the "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" section in the guide. For information about database backup, see the *Oracle Database Backup and Recovery User's Guide*.

To back up the installation at this point:

1. Back up the web tier on WEBHOST1:
   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 Oracle instance on the web tier using the following command:

      ```
      tar -cvpf BACKUP_LOCATION/web_instance_name.tar ORACLE_INSTANCE
      ```

   d. Start the instance using `opmnctl`:

      ```
      cd 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 operating system tools such as tar for cold backups if possible.
3. Back up the Administration Server and Managed Server domain directory to save your domain configuration. The configuration files all exist in the *ORACLE_BASE*`/admin/`*domain_name* directory. Run the following command on SOAHOST1 to create the backup:

   ```
   tar -cvpf edgdomainback.tar ORACLE_BASE/admin/domain_name
   ```

11

Extending the Domain to Include Imaging

This chapter describes how to extend a domain with Oracle WebCenter Content: Imaging using the Oracle Fusion Middleware Configuration Wizard. It contains the following sections:

- Section 11.24, "Configuring Server Migration for the WLS_IMG Managed Servers"
- Section 11.25, "Backing Up the Installation"

Note: Before starting the setup process, read the *Oracle Fusion Middleware Release Notes* for your platform for additional installation and deployment information.

11.1 Overview of Extending the Domain to Include Imaging

The Oracle WebCenter Content: Imaging system is installed using the WL_HOME and ORACLE_HOME locations created in Chapter 6, "Installing the Software for an Enterprise Deployment," on a shared storage. WCCHOST1 and WCCHOST2 mount MW_HOME and reuse the existing Oracle WebLogic Server, Oracle SOA Suite, and Oracle WebCenter Content binary installations. The `pack` and `unpack` utilities are used to bootstrap the domain configuration for the WLS_IMG1 and WLS_IMG2 servers in these two new nodes. As a result, you do not need to install any software in these two nodes. For the Imaging system to work properly, WCCHOST1 and WCCHOST2 must maintain the same system requirements and configuration that was required for installing Oracle Fusion Middleware in SOAHOST1 and SOAHOST2. Otherwise, unpredictable behavior in the execution of binaries may occur.

Extend the domain to include Imaging. Table 11–1 lists the steps for configuring Imaging and other tasks required for extending the domain with Imaging Managed Servers.

Table 11–1 Steps for Extending the Domain with Imaging

Step	Description	More Information
Prepare for extending the domain for Imaging	Enable a VIP mapping for each of the host names.	Section 11.2, "Enabling VIP4 and VIP5 in WCCHOST1and WCCHOST2"
Extend the domain for Imaging	Extend the domain you created in Chapter 8, "Creating a Domain for an Enterprise Deployment."	Section 11.3, "Extending the Domain for Imaging"
Disable host name verification for Imaging	Disable host name verification while setting up and validating the topology.	Section 11.4, "Disabling Host Name Verification for the WLS_IMG Managed Servers"
Propagate the domain configuration to the Imaging Managed Servers	Propagate the start scripts and classpath configuration from the Administration Server's domain directory to the Managed Server domain directories.	Section 11.5, "Propagating the Domain Configuration to the Managed Server Domain Directories"
Configure a JMS persistence store	Configure the location for the JMS persistence stores as a directory that is visible to both Imaging Managed Servers.	Section 11.6, "Configuring a JMS Persistence Store for Imaging"
Configure a default persistence store	Configure a default persistence store for transaction recovery.	Section 11.7, "Configuring a Default Persistence Store for Transaction Recovery"
Restart the Administration Server for the domain	Stop and then restart the Administration Server.	Section 11.8, "Restarting the Administration Server"
Start the Imaging Managed Servers	Start the WLS_IMG1 and WLS_IMG2 Managed Servers.	Section 11.9, "Starting the Imaging Managed Servers"

Table 11–1 (Cont.) Steps for Extending the Domain with Imaging

Step	Description	More Information
Verify the configuration of GridLink data sources and Oracle Notification Service (ONS)	Follow these instructions to verify that the configuration of GridLink data sources and ONS is correct.	Section 11.10, "Validating GridLink Data Sources for Imaging"
Configure System MBeans for Imaging	Configure the `InputDirectories`, `SampleDirectory`, and `GDFontPath` MBeans in the System MBean Browser.	Section 11.12, "Configuring System MBeans for Imaging"
Enable the Imaging feature set in WebCenter Content	Enable the **IpmRepository** component of Content Server.	Section 11.13, "Enabling the Imaging Feature Set in Oracle WebCenter Content"
Configure the Imaging Viewer Cache	Set the Imaging viewer to use cached documents.	Section 11.14, "Configuring the Imaging Viewer Cache"
Encrypt cached documents	Encrypt cached documents if additional security is required.	Section 11.15, "Encrypting Cached Documents"
Add the listen addresses for the Imaging Managed Servers	Add the host names of the WLS_IMG1 and WLS_IMG2 Managed Servers (WCCHOST1VHN1 and WCCHOST2VHN1) to the `SocketHostNameSecurityFilter` parameter list.	Section 11.16, "Adding the Imaging Server Listen Addresses to the List of Allowed Hosts in Oracle WebCenter Content"
Create a connection between the Imaging and WebCenter Content Managed Servers	Create a connection to Oracle WebCenter Content Server.	Section 11.17, "Creating a Connection to Oracle WebCenter Content Server"
Configure the BPEL CSF credentials	Configure the required credentials to communicate with Oracle SOA Suite	Section 11.18, "Configuring BPEL CSF Credentials"
Configure a workflow connection for Imaging	Create and test a workflow connection.	Section 11.19, "Configuring a Workflow Connection"
Configure Oracle HTTP Server with the extended domain	Configure the Oracle HTTP Server with the Managed Servers, and set the WLS Cluster address for IMG_Cluster.	Section 11.20, "Configuring Oracle HTTP Server for the WLS_IMG Managed Servers"
Set the front-end HTTP host and port for Imaging.	Configure the front-end HTTP host and port for IMG_Cluster.	Section 11.21, "Setting the Front-End HTTP Host and Port"
Validate access to WebCenter Content through Oracle HTTP Server	Verify the URLs to ensure that appropriate routing and failover is working from Oracle HTTP Server to IMG_Cluster.	Section 11.22, "Validating Access Through the Load Balancer"
Configure the Imaging Managed Servers to use custom key stores	Configure Node Manager with the custom key stores for Imaging.	Section 11.23, "Configuring Node Manager for the WLS_IMG Managed Servers"
Configure server migration for the Imaging Managed Servers.	Specify the Imaging Managed Server names, host names, and cluster name for migration.	Section 11.24, "Configuring Server Migration for the WLS_IMG Managed Servers"
Back up the Imaging configuration	Back up the newly extended domain configuration.	Section 11.25, "Backing Up the Installation"

11.2 Enabling VIP4 and VIP5 in WCCHOST1and WCCHOST2

The Oracle WebCenter Content: Imaging system uses virtual host names as the listen addresses for the Managed Servers on which Imaging is running. These virtual host names and corresponding virtual IPs are required to enable server migration for the Oracle WebCenter Content: Imaging component. You must enable a virtual IP (VIP4 or

VIP5) mapping to WCCHOST1VHN1 on WCCHOST1 and WCCHOST2VHN1 on WCCHOST2, and you must also correctly resolve the host names in the network system used by the topology (by either DNS Server or `/etc/hosts` resolution).

For information about how to enable the VIPs, see Section 3.5, "Enabling Virtual IP Addresses for an Enterprise Deployment."

11.3 Extending the Domain for Imaging

You extend the domain configured in Chapter 10, "Extending the Domain to Include Oracle WebCenter Content," to include Oracle WebCenter Content: Imaging. The instructions in this section assume that the Imaging deployment uses the same database service as the Oracle WebCenter Content deployment (`wccedg.mycompany.com`). However, a deployment can choose to use a different database service specifically for Imaging.

Note: Before performing these steps, back up the domain as described in the *Oracle Fusion Middleware Administrator's Guide*.

To extend the domain for Imaging:

1. Make sure that the database where you installed the repository is running.

   For Oracle RAC databases, it is recommended that all instances are running, so that the validation check later on becomes more reliable.

2. On SOAHOST1, change the directory to the location of the Fusion Middleware Configuration Wizard. This is within the Oracle Common home directory (domain extensions are run from the node where the Administration Server resides).

   ```
   cd ORACLE_COMMON_HOME/common/bin
   ```

3. Start the Fusion Middleware Configuration Wizard:

   ```
   ./config.sh
   ```

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

5. In the Select a WebLogic Domain Directory screen, select the WebLogic Server domain directory (*ORACLE_BASE*`/admin/`*domain_name*`/aserver/`*domain_name*), and click **Next**.

6. In the Select Extension Source screen (Figure 11–1), make these selections:

   - Select **Extend my domain automatically to support the following added products**.
   - Select these products:

     Oracle WebCenter Content: Imaging Viewer Cache - 11.1.1.0 [wcc]

     Oracle WebCenter Content: Imaging - 11.1.1.0 [wcc]

     (If you select one of the preceding products, the other one is automatically selected.)

     Notes:

     - AXF for BPEL is included in Imaging.
     - For information about including Oracle WebCenter Content: AXF for BPM in Imaging, see the *Oracle WebCenter Content Installation Guide*.

     The following products should already be selected and grayed out. They were selected when you created the domain (Section 8.3) or extended it for Oracle SOA Suite components (Section 9.3) or WebCenter Content (Section 10.2).

     Basic WebLogic Server Domain - 10.3.6.0 [wlserver_10.3]

     Oracle SOA Suite for developers - 11.1.1.0 [soa]

     Oracle SOA Suite - 11.1.1.0 [soa]

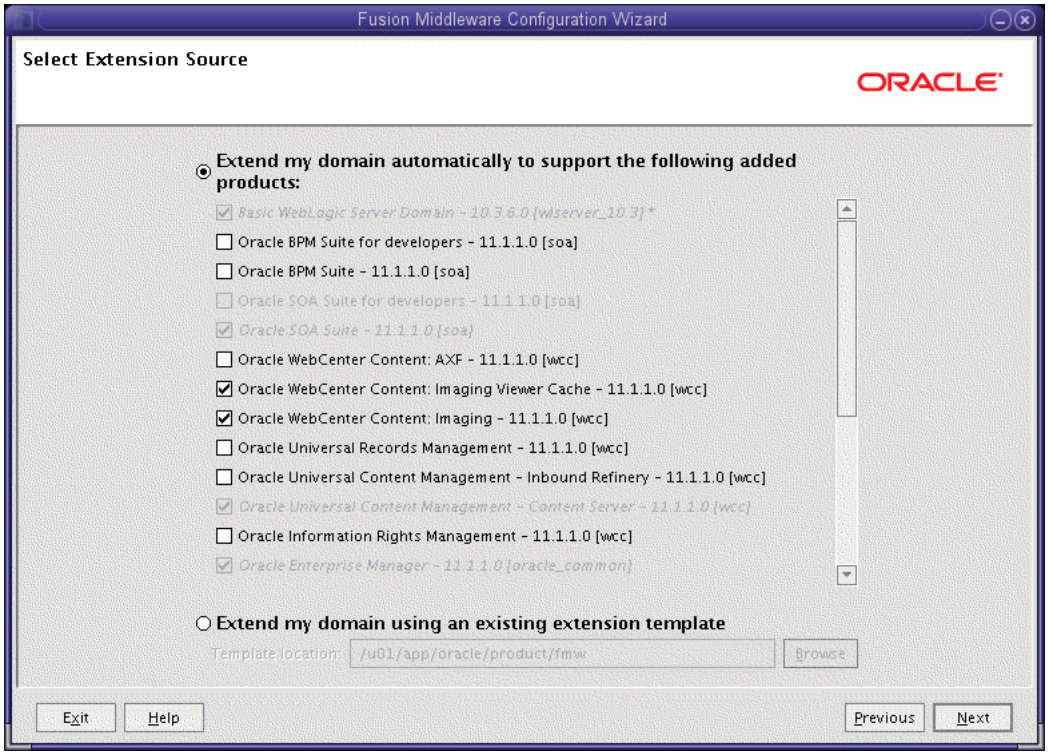
     Oracle Universal Content Management - Content Server - 11.1.1.0 [wcc]

     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]

Figure 11–1 Select Extension Source Screen for Imaging



Click **Next**.

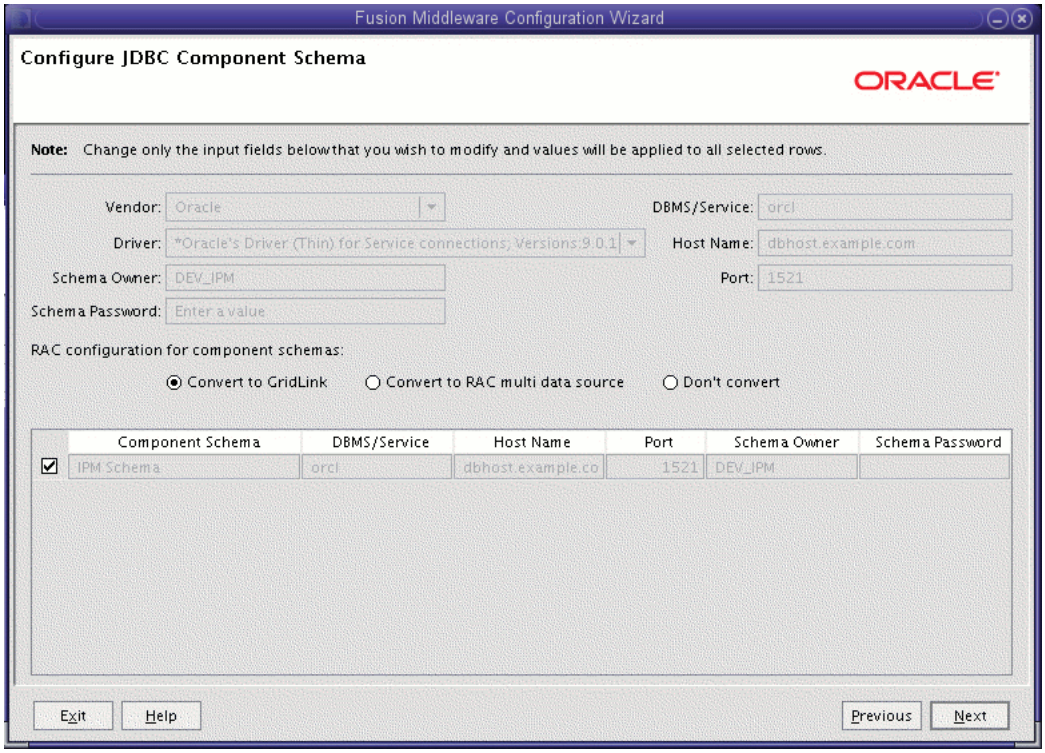
7. In the Configure JDBC Component Schema screen, which Figure 11–2 shows, do the following:

   a. Select **IPM Schema** only (for Imaging). Do not select any of the other existing schemas.

   b. For the RAC configuration, you can select **Convert to GridLink** or **Convert to RAC multi data source** (described in Appendix A, "Using Multi Data Sources with Oracle RAC"). For the instructions given here, select **Convert to GridLink**.

      After you select a RAC configuration, all selected schemas are grayed.

Figure 11–2 Configure JDBC Component Schema Screen for Imaging

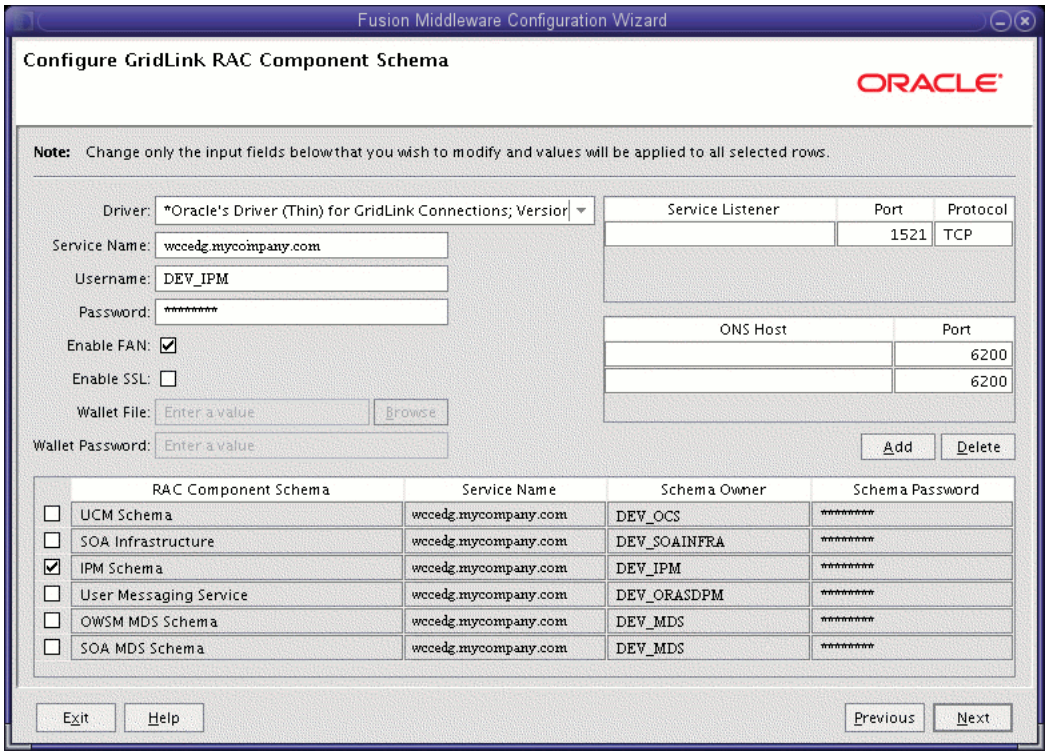


c. Click **Next**.

8. In the Configure GridLink RAC Component Schema screen (Figure 11–3), do the following:

   a. Select **IPM Schema** (for Imaging). Leave the other data sources as they are.

Figure 11–3 Configure GridLink RAC Component Schema Screen for Imaging



   b. Enter values for the following fields, specifying the connection information for the GridLink RAC database that was seeded through RCU:

      – **Driver:** Select **Oracle driver (Thin) for GridLinkConnections; Versions:10 and later**.

      – **Service Name:** Enter the service name of the Oracle RAC database in lowercase letters, followed by the domain name; for example, `wccedg.mycompany.com`.

      – **Username:** Enter the complete user name for the database schema owner of the corresponding component.

        This book uses `DEV` as the prefix of user names for the database schemas.

      – **Password:** Enter the password for the database schema owner.

      – Select **Enable FAN**.

      – **Enable SSL:** Leave this option deselected.

        If you select SSL to enable Oracle Notification Service (ONS) notification encryption, provide the appropriate **Wallet File** and **Wallet Password** details.

      – **Service listener:** Enter the Oracle Single Client Access Name (SCAN) address and port for the Oracle RAC database being used. The protocol should be `TCP`.

Oracle recommends that you use a SCAN address to specify the Service Listener (and OSN Host) so you do not need to update a GridLink data source containing a SCAN address if you add or remove Oracle RAC nodes. To determine the SCAN address, query the `remote_listener` parameter in the database:

```
SQL>show parameter remote_listener;

NAME              TYPE        VALUE
-----             ------      -------
remote_listener   string      db-scan.mycompany.com :1521
```

Note: For Oracle Database 11*g* Release 1 (11.1), use the virtual IP and port of each database instance listener; for example:

```
custdbhost1-vip.mycompany.com (port 1521)
```

and

```
custdbhost2-vip.mycompany.com (1521)
```

For Oracle Database 10*g*, use multi data sources to connect to an Oracle RAC database. For information about configuring multi data sources, see Appendix A, "Using Multi Data Sources with Oracle RAC."

– **ONS Host:** Enter here also the SCAN address for the RAC database and the ONS remote port, as reported by the database:

```
[orcl@CUSTDBHOST1 ~]$ srvctl config nodeapps -s
ONS exists: Local port 6100, remote port 6200, EM port 2016
```

Note: For Oracle Database 11*g* Release 1 (11.1), use the host name and port of each database's ONS service; for example:

```
custdbhost1.mycompany.com (port 6200)
```

and

```
custdbhost2.mycompany.com (6200)
```

---

c. Click **Next**.

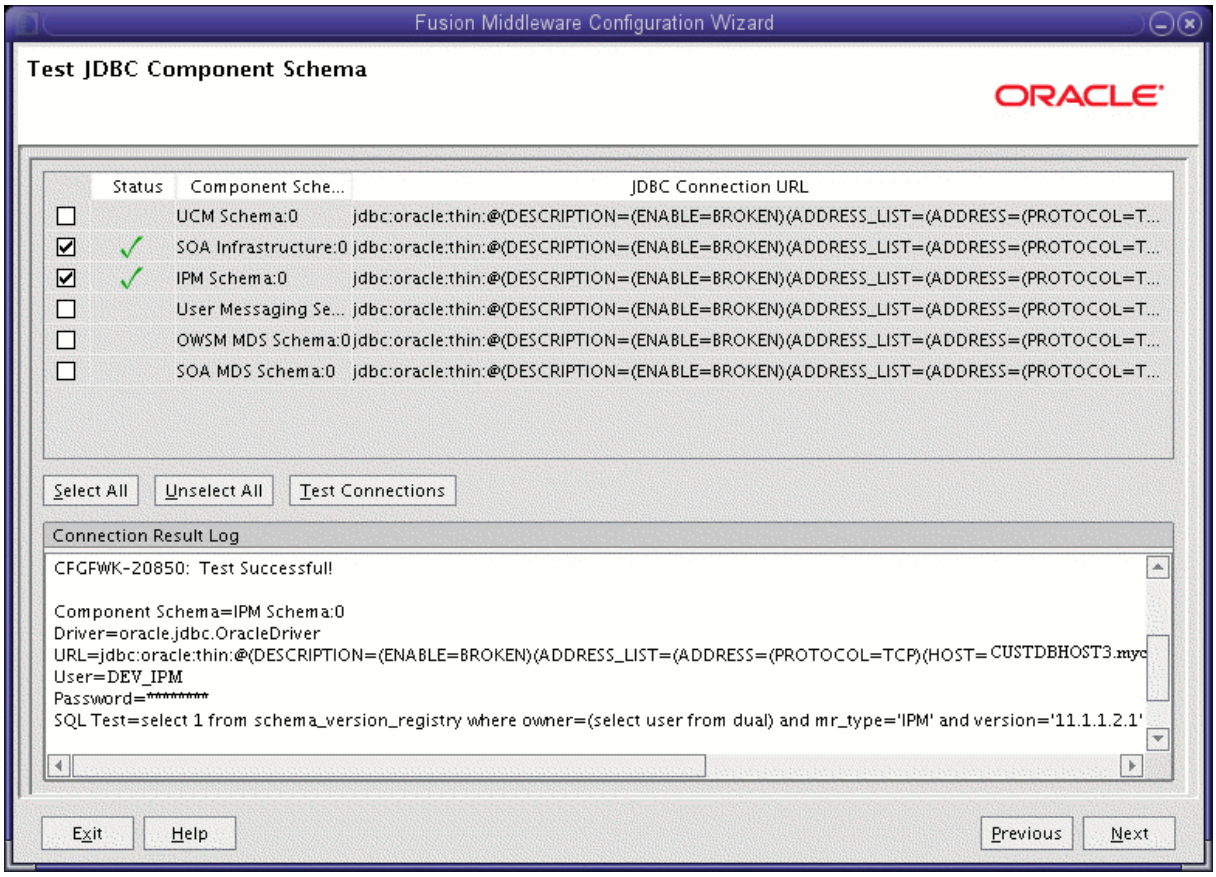
Note: Leave the **UCM Schema**, **SOA Infrastructure**, **User Messaging Service**, **OWSM MDS Schema**, and **SOA MDS Schema** information as is.

---

9. In the Test JDBC Component Schema screen, select the **IPM Schema** row, then click **Test Connections**.

   The **Connection Results Log** displays the results. Ensure that the connection to the database that contains the schema was successful. If not, click **Previous** to return to the previous screen, correct your entry, and then retry the test.

Figure 11–4 Test JDBC Component Schema Screen for Imaging



   Click **Next** when the connection is successful.

10. In the Optional Configuration screen, select the following options:

    - **JMS Distributed Destination**
    - **Managed Servers, Clusters and Machines**
    - **Deployment and Services**

    Click **Next**.

11. In the Select JMS Distributed Destination Type screen, select UDD from the drop-down list for the JMS modules of all Oracle Fusion Middleware components. Click **Next**.

    If an override warning appears, click **OK** to acknowledge it.

12. In the Configure Managed Servers screen, add the required Managed Servers.

    A server is created automatically. Rename this to WLS_IMG1 and add a new server called WLS_IMG2. Give these servers the attributes listed in Table 11–2. Do not modify the other servers that appear in this screen; leave them as they are.

Table 11–2 Managed Servers

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled
WLS_IMG1	WCCHOST1VHN1	16000	n/a	No
WLS_IMG2	WCCHOST2VHN2	16000	n/a	No

Click **Next**.

13. In the Configure Clusters screen, click **Add** to add the clusters as shown in Table 11–3. Do not modify the other clusters that appear in this screen; leave them as they are.

Table 11–3 Clusters

Name	Cluster Messaging Mode	Multicast Address	Multicast Port	Cluster Address
IMG_Cluster	unicast	n/a	n/a	Leave empty

Click **Next**.

14. In the Assign Servers to Clusters screen, add the following. Do not modify the other assignments that appear in this screen; leave them as they are.
    - **IMG_Cluster:**
      - WLS_IMG1
      - WLS_IMG2

    Click **Next**.

15. In the Configure Machines screen, open the **Unix Machine** tab. You should see the WCCHOST1 and WCCHOST2 machines and have the following entries:

Table 11–4 Machines

Name	Node Manager Listen Address
SOAHOST1	SOAHOST1
SOAHOST2	SOAHOST2
ADMINVHN	localhost
WCCHOST1	WCCHOST1
WCCHOST2	WCCHOST2

Leave all other fields to their default values. Click **Next**.

16. In the Assign Servers to Machines screen, assign servers to machines as follows:
    - Assign **WLS_IMG1** to **WCCHOST1**.
    - Assign **WLS_IMG2** to **WCCHOST2**.

    Click **Next**.

17. 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.)
    - **WSM-PM** should be targeted only to **SOA_Cluster**.
    - The **oracle.rules***, **oracle.sdp.*** and **oracle.soa.*** deployments should be targeted to **SOA_Cluster** only, except for the **oracle.soa.workflow.wc** library, which should be targeted to both the **SOA_Cluster** and the **IMG_Cluster**.

    Click **Next**.
18. In the Target Service to Cluster or Servers screen, click **Next**.
19. In the Configuration Summary screen, click **Extend**.
20. If a dialog window appears warning about conflicts in ports for the domain, click **OK**. This should be due to pre-existing servers in the nodes and the warning can be ignored.
21. In the Creating Domain screen, click **Done**.
22. Restart the Administration Server to make these changes to take effect.

    To restart the Administration Server, stop it first using the Administration Console and then start it again as described in Section 8.4.3, "Starting the Administration Server on SOAHOST1."

11.4 Disabling Host Name Verification for the WLS_IMG Managed Servers

For the enterprise deployment described in this guide, you set up the appropriate certificates to authenticate the different nodes with the Administration Server after you have completed the procedures to extend the domain for Imaging. You must disable the host name verification for the WLS_IMG1 and WLS_IMG2 Managed Servers to avoid errors when managing the different WebLogic Server instances. For more information, see Section 8.4.5, "Disabling Host Name Verification."

You enable host name verification again once the enterprise deployment topology configuration is complete. For more information, see Section 13.3, "Enabling Host Name Verification Certificates for Node Manager."

11.5 Propagating the Domain Configuration to the Managed Server Domain Directories

To propagate the domain configuration:

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

```
cd ORACLE_COMMON_HOME/common/bin

./pack.sh -managed=true -domain=ORACLE_BASE/admin/domain_name/aserver/domain_
name -template=edgdomaintemplateIMG.jar -template_name=edgdomain_templateIMG
```

2. Run the following command on SOAHOST1 to copy the template pack created in the previous step to WCCHOST2:

> **Note:** Assuming that WCCHOST1 shares the ORACLE_HOME with SOAHOST1, the template will be present in the same directory in WCCHOST1; otherwise, copy it also to WCCHOST1.

```
scp edgdomaintemplateIMG.jar oracle@WCCHOST2:ORACLE_BASE/product/fmw/oracle_
common/common/bin
```

3. Run the `unpack` command on WCCHOST1 to unpack the propagated template.

> **Note:** Make sure to run `unpack` from the *ORACLE_COMMON_HOME*`/common/bin/` directory, not from *WL_HOME*`/common/bin/`.

```
cd ORACLE_COMMON_HOME/common/bin

./unpack.sh -domain=ORACLE_BASE/admin/domain_name/mserver/domain_name
-template=edgdomaintemplateIMG.jar -app_dir= ORACLE_BASE/admin/domain_
name/mserver/applications -overwrite_domain=true
```

> **Note:** The *ORACLE_BASE*`/admin/`*domain_name*`/mserver/` directory must exist before you run `unpack`. In addition, the *ORACLE_BASE*`/admin/`*domain_name*`/mserver/applications/` directory must be empty.

4. Repeat step 3 for WCCHOST2.

11.6 Configuring a JMS Persistence Store for Imaging

Configure the location for the JMS persistence stores as a directory that is visible from both nodes. By default, the JMS servers used by Oracle WebCenter Content: Imaging are configured with no persistence store and use the WebLogic Server store (*ORACLE_BASE*`/admin/`*domain_name*`/mserver/`*domain_name*`/servers/`*server_name*`/data/store/` default). You must change the Imaging JMS server persistence store to use a shared base directory as follows:

1. Log in to the WebLogic Server Administration Console.
2. In the **Domain Structure** tree on the left, expand the **Services** node, and then click the **Persistence Stores** node.

3. On the Summary of Persistence Stores page, click **Lock & Edit**.
4. Click **New**, and then **Create File Store**.
5. Enter a **name** (for example, `IMGJMSServer1Store`, which allows you identify the service it is created for) and target WLS_IMG1. Enter a **directory** that is located in shared storage so that it is accessible from both WCCHOST1 and WCCHOST2 (*`ORACLE_BASE`*`/admin/`*`domain_name`*`/`*`img_cluster_name`*`/jms`).
6. Click **OK** and activate the changes.
7. In the **Domain Structure** tree on the left, expand the **Services** node, and then click the **Messaging->JMS Servers** node.
8. On the Summary of JMS Servers page, click the `IpmJmsServer1 JMS Server` (represented as a hyperlink) in the **Name** column of the table.
9. On the settings page for the JMS server, click **Lock & Edit**.
10. In the Persistent Store drop-down list, select **IMGJMSServer1Store**.
11. Click **Save and Activate**.
12. Repeat the steps and create **IMGJMSServer2Store** for IMGJMSServer2.

11.7 Configuring a Default Persistence Store for Transaction Recovery

Each server has a transaction log which stores information about committed transactions that are coordinated by the server that may not have been completed. 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 the server.

> **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 for WLS_IMG1:

1. Log in to the WebLogic Server Administration Console.
2. In the **Domain Structure** tree on the left, expand the **Environment** node, and then click the **Servers** node.
3. On the Summary of Servers page, click WLS_IMG1 (represented as a hyperlink) in the **Name** column of the table. The settings page for the WLS_IMG1 server opens with the **Configuration** tab active.
4. Open the **Services** tab.
5. Click **Lock & Edit**.
6. 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`*`/`*`img_cluster_name`*`/tlogs`

7. Click **Save** and activate the changes.
8. Repeat the step for the WLS_IMG2 server.

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 WCCHOST1 and WCCHOST2 must be able to access this directory. This directory must also exist before you restart the server.

11.8 Restarting the Administration Server

Restart the Administration Server to make these changes take effect. To restart the Administration Server, stop it first using the Administration Console and then start it again as described in Section 8.4.3, "Starting the Administration Server on SOAHOST1."

11.9 Starting the Imaging Managed Servers

To start the WLS_IMG1 Managed Server on WCCHOST1:

1. Start the WLS_IMG1 Managed Server:
   a. Log in to the WebLogic Server Administration Console at `http://ADMINVHN:7001/console`.
   b. In the **Domain Structure** tree on the left, expand the **Environment** node, and then select **Servers**.
   c. On the Summary of Servers page, open the **Control** tab.
   d. Select WLS_IMG1 from the **Servers** column of the table.
   e. Click **Start**.
2. Access `http://WCCHOST1VHN1:16000/imaging` to verify the status of WLS_IMG1. The Oracle WebCenter Content: Imaging login page appears. Enter your WebLogic Server administration user name and password to log in.

   Verify that the PROCESSES initialization parameter for the database is set to a high enough value. For details, see Section 5.2.3, "Initialization Parameters." This error often occurs when you start servers that are subsequent to the first server.
3. Start the WLS_IMG2 Managed Server:
   a. Log in to the WebLogic Server Administration Console at `http://ADMINVHN:7001/console`.
   b. In the **Domain Structure** tree on the left, expand the **Environment** node, and then select **Servers**.
   c. On the Summary of Servers page, open the **Control** tab.
   d. Select WLS_IMG2 from the **Servers** column of the table.
   e. Click **Start**.
4. Access `http://WCCHOST2VHN1:16000/imaging` to verify the status of WLS_IMG2. The Oracle WebCenter Content: Imaging login page appears. Enter your WebLogic Server administration user name and password to log in.

Note: These instructions assume that the host name verification displayed previously for the Oracle WSM or Oracle SOA Suite Managed Servers in SOAHOST2 and that the Node Manager is already running on SOAHOST2.

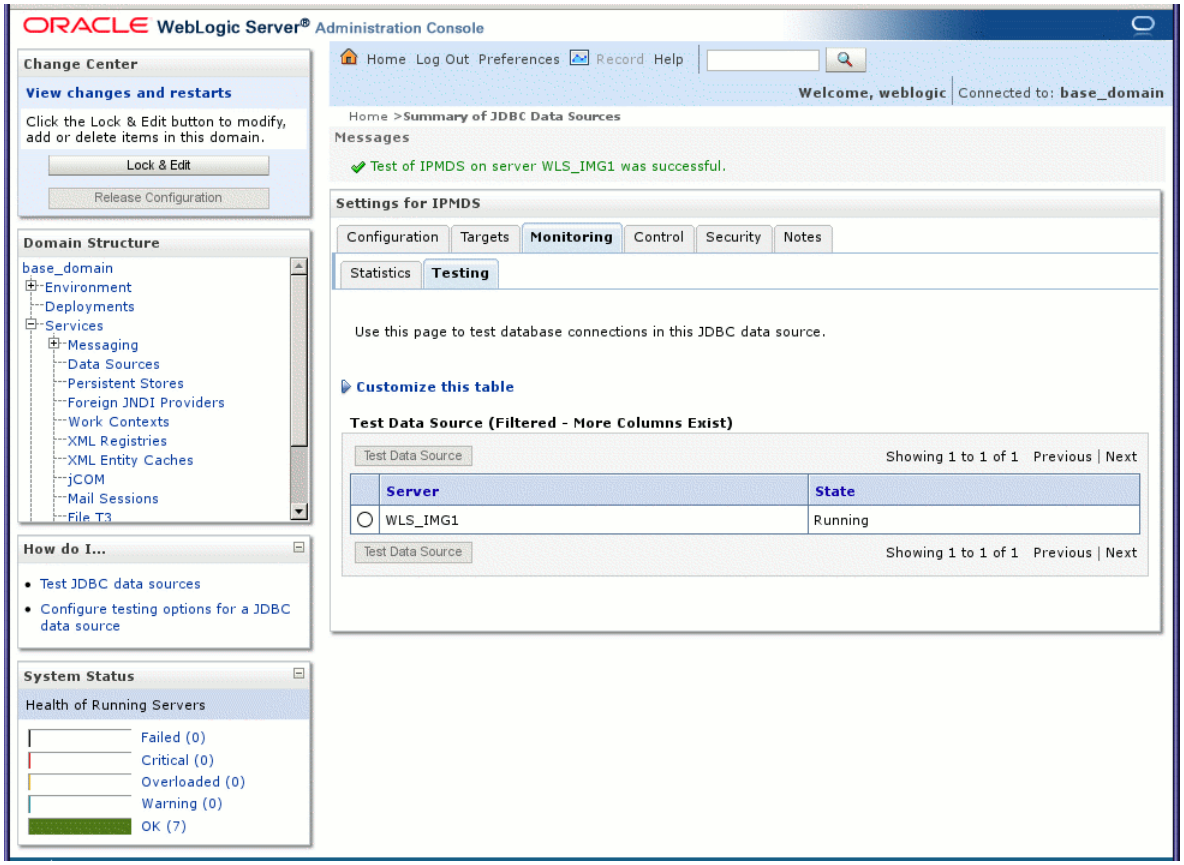
11.10 Validating GridLink Data Sources for Imaging

After the servers are started, verify that the GridLink data sources are correctly configured and that the ONS setup is correct. Perform this procedure for every GridLink data source created.

To verify the configuration of a GridLink data source for Imaging:

1. Log in to the WebLogic Server Administration Console.
2. In the **Domain Structure** tree, expand **Services**, then click **Data Sources**.
3. Click the name of a GridLink data source that was created.
4. Click the **Monitoring** tab.
5. Click the **Testing** tab (Figure 11–5), select one of the servers, and click **Test Data Source**.

Figure 11–5 Testing a GridLink Data Source for Imaging



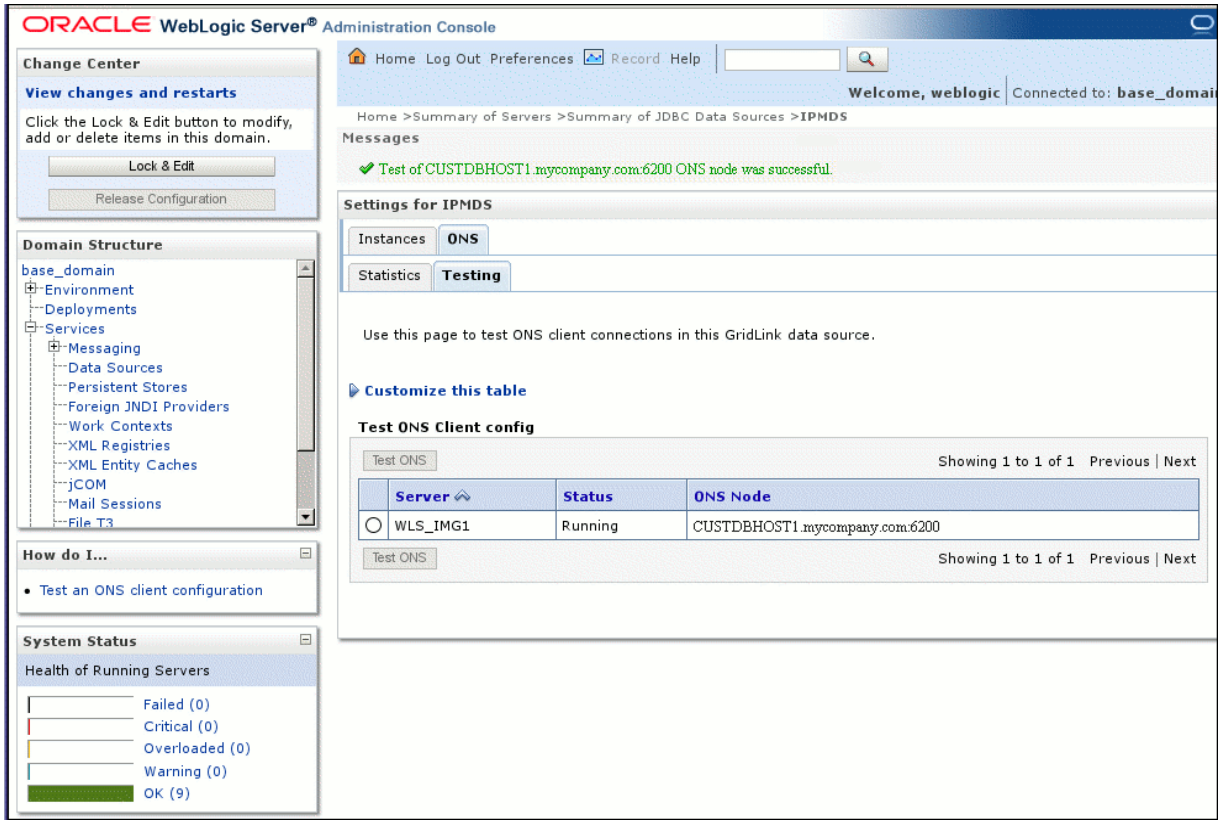
The test should be successful if the configuration is correct.

6. Repeat the test for every WebLogic Server instance that uses the GridLink data source.

To verify the configuration of ONS for a GridLink data source for Imaging:

1. Log in to the WebLogic Server Administration Console.
2. In the **Domain Structure** tree, expand **Services**, then click **Data Sources**.
3. Click the name of a GridLink data source.
4. Click the **Monitoring** tab.
5. Click the **ONS** tab and then the **Testing** tab (Figure 11–6).
6. Select a server, and click **Test ONS**.

Figure 11–6 Testing the ONS Configuration for Imaging



The test should be successful if the configuration is correct. If the ONS test fails, verify that the ONS service is running in the Oracle RAC database nodes:

```
[orcl@CUSTDBHOST1 ~]$ srvctl status scan_listener
SCAN Listener LISTENER_SCAN1 is enabled
SCAN listener LISTENER_SCAN1 is running on node CUSTDBHOST1
SCAN Listener LISTENER_SCAN2 is enabled
SCAN listener LISTENER_SCAN2 is running on node CUSTDBHOST2
SCAN Listener LISTENER_SCAN3 is enabled
SCAN listener LISTENER_SCAN3 is running on node CUSTDBHOST2


[orcl@CUSTDBHOST1 ~]$ srvctl config nodeapps -s
ONS exists: Local port 6100, remote port 6200, EM port 2016


[orcl@CUSTDBHOST1 ~]$ srvctl status nodeapps | grep ONS
ONS is enabled
ONS daemon is running on node: CUSTDBHOST1
ONS daemon is running on node: CUSTDBHOST2
```

7. Repeat the ONS test for every WebLogic Server instance that uses the GridLink data source.

11.11 Validating Deployment of the Imaging Viewer Cache

After the Imaging servers are started, verify that the Viewer Cache was deployed correctly.

To validate deployment of the Imaging viewer cache:

1. In the WebLogic Server Administration Console, click **Deployments** under Domain Structure on the left.
2. In the `imaging-vc` row of the **Deployments** table, confirm that the **State** value is `Active` and the **Health** value is `OK`.

   If the **State** or **Health** value is different for `imaging-vc`, you need to redeploy the feature before proceeding.

11.12 Configuring System MBeans for Imaging

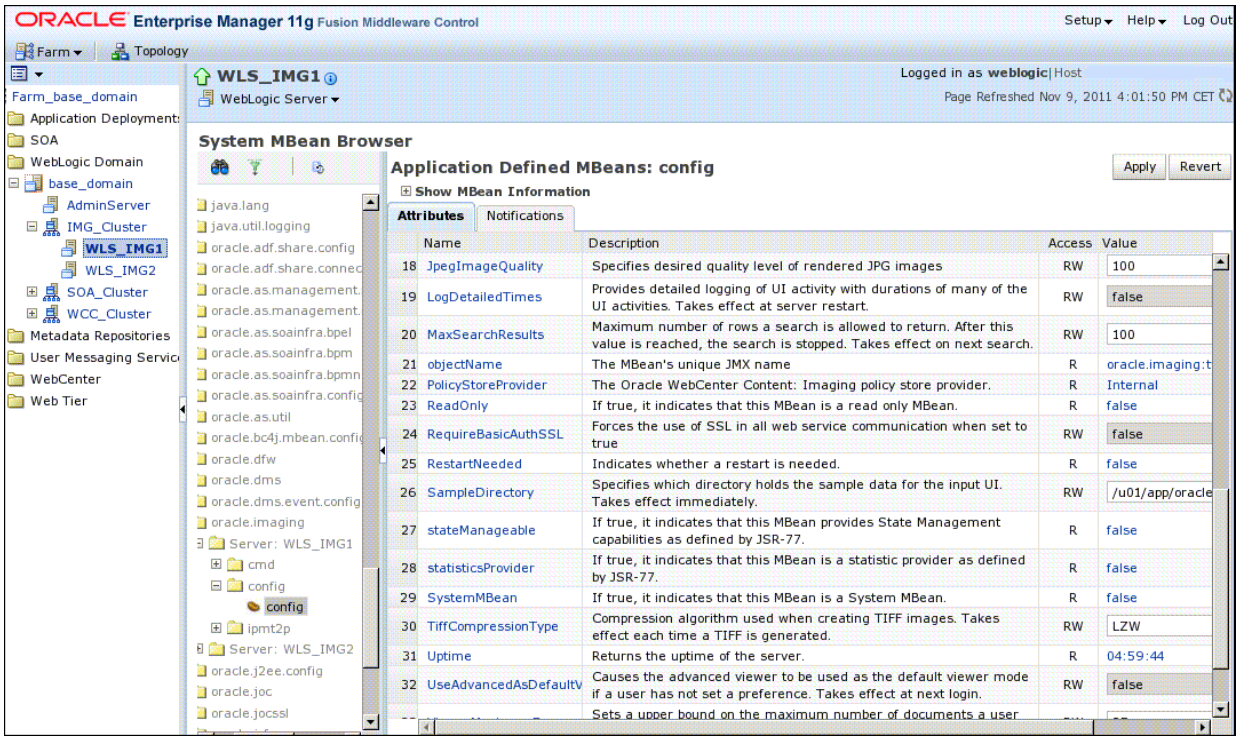
You can configure the following system MBeans for Imaging:

- `InputDirectories`
- `SampleDirectory`
- `GDFontPath`

To configure system MBeans for Imaging:

1. Log in to Oracle Enterprise Manager Fusion Middleware Control at `http://ADMINVHN:7001/em` (Figure 11–7).

Figure 11–7 System MBean Browser



2. In the navigation tree on the left, expand the farm domain name, then **WebLogic Domain**, then the domain name, and then **IMG_Cluster**, and then click the **WLS_IMG1** link.
3. At the top, click the WebLogic Server drop-down menu, and choose **System MBean Browser**.
4. Expand **Application Defined MBeans** and then **oracle.imaging**.
5. Expand **Server: WLS_IMG1** and then **config**.
6. Click the **config** bean link.
7. In the right pane, set the **InputDirectories** MBean to specify the path to the input files: *ORACLE_BASE*`/admin/`*domain_name*`/`*img_cluster_name*`/input_files`.

   All Oracle WebCenter Content Servers involved must be able to resolve this location (that is, through the NFS mount point).
8. Set the **SampleDirectory** MBean: *ORACLE_BASE*`/admin/`*domain_name*`/`*img_cluster_name*`/input_files/Samples`.

   To process input files, the input agent must have the appropriate permissions for the input directory, and the input directory must allow file locking. The input agent requires that the user account that is running the WebLogic Server service have read and write privileges for the input directory and for all files and subdirectories in the input directory. These privileges are required so that the input agent can move the files to the various directories as it works on them. File locking on the share is needed by the input agent to coordinate actions between servers in the cluster.

9. Set the **GDFontPath** MBean to specify the path to the GD fonts for the X Windows environment. Check with your system administrator. The default is likely `/usr/share/X11/fonts/TTF` or `/usr/lib/X11/fonts/TTF`.
10. Click **Apply**.

11.13 Enabling the Imaging Feature Set in Oracle WebCenter Content

To enable the Imaging feature set in Oracle WebCenter Content:

1. Log in to Oracle Content Server at `http://WCCHOST1:16200/cs`.
2. Open the **Administration** tray or menu, and choose **Admin Server**.
3. Select **Integration** components.
4. On the Component Manager page, enable the **IpmRepository** component.
5. Click **Update** and confirm the action.
6. Restart the Managed Server, and then restart all Managed Servers in the Oracle WebCenter Content cluster.

11.14 Configuring the Imaging Viewer Cache

The Imaging viewer can cache documents on the server outside of the repository to increase rendering speed on the client machine. Security for the cached documents is controlled by authentication for the server on which they are stored. If the server is considered secure, no additional security is necessary.

If additional security is required, you can encrypt cached documents, as described in Section 11.15, "Encrypting Cached Documents." For information about when to use the precache option and how to optimize ingestion and rendering when processing a large number of documents, see "Balancing Ingestion and Rendering When Viewer Cache is Enabled" in the *Oracle WebCenter Content Administrator's Guide for Imaging*.

To set the Imaging viewer to use cached documents, the following system MBeans need to be set:

- `ViewerCachePath`
- `ViewerCacheDays`
- `ViewerCacheEnablePrecache`

To configure these MBeans for the Imaging Viewer Cache, use the method described in Section 11.12, "Configuring System MBeans for Imaging," as follows:

- Set the **ViewerCachePath** MBean to the location where documents should be cached:

  ```
  ORACLE_BASE/admin/domain_name/img_cluster_name/ViewerCache
  ```

> **Note:** The **ViewerCachePath** MBean should be set to a location available to all servers in the cluster. If the directory path is not available to all servers, then each server will cache documents locally, resulting in multiple instances of the entire cache.

- Set the **ViewerCacheDays** MBean to `30`.

> **Note:** This configuration value specifies the number of days cached images should be retained before they are purged from the cache. Setting **ViewerCacheDays** equal to `0` prevents the cache from being purged.

- Set the **ViewerCacheEnablePrecache** MBean to `true`.

> **Note:** This configuration value specifies whether documents should be cached as soon as they are ingested into Imaging (precached).

For information about moving the viewer cache to a new location, see "Changing the Viewer Cache Path" in the *Oracle WebCenter Content Installation Guide*.

11.15 Encrypting Cached Documents

If additional security is required, Imaging can be configured to encrypt cached documents. Encryption makes additional processing necessary to decrypt a document for viewing and reduces rendering speed. Even if Imaging is configured to encrypt the cached documents, there is a brief period of time during caching when generated documents are not encrypted.

To enable encryption of cached documents, add a new password credential to the domain through Oracle Enterprise Manager Fusion Middleware Control, and set the **ViewerCacheEnableEncryption** MBean:

1. Log in to Fusion Middleware Control at `http://ADMINVHN:7001/em`(Figure 11–7).
2. Select the WebLogic Server domain for Oracle WebCenter Content.
3. From the **WebLogic Domain** menu, select **Security** and then **Credentials**.
4. Select the map **oracle.imaging**. If no map named `oracle.imaging` exists, click **Create Map**, enter **oracle.imaging** for the map name, and then select it.
5. Click **Create Key**. Name the key **viewer.cache**, and select the type **Password**.
6. Enter a user name. The user name does not need to exist in any system.
7. Enter a password, confirm it, and then click **OK**.
8. Set the **ViewerCacheEnableEncryption** MBean to `true`, using the method described in Section 11.12, "Configuring System MBeans for Imaging."

> **Note:** This is an additional option to encrypt the page images in the cache. The password credential must exist on the domain before you set the **ViewerCacheEnableEncryption** MBean. To add a password credential, use the method described in Section 11.15, "Encrypting Cached Documents."

For information about disabling encryption, see "Disabling Encryption of Cached Documents" in the *Oracle WebCenter Content Installation Guide*.

11.16 Adding the Imaging Server Listen Addresses to the List of Allowed Hosts in Oracle WebCenter Content

To add the host names of the WLS_IMG1 and WLS_IMG2 Managed Servers (WCCHOST1VHN1 and WCCHOST2VHN1, respectively) to the `SocketHostNameSecurityFilter` parameter list:

1. Open the file *`ORACLE_BASE`*`/admin/`*`domain_name`*`/`*`wcc_cluster_name`*`/cs/config/config.cfg` in a text editor.
2. Remove or comment out the following line:

   ```
   SocketHostAddressSecurityFilter=127.0.0.1|WCCHOST1|WCCHOST2|WEBHOST1|WEBHOST2
   ```

3. Add the following two lines to include the WLS_IMG1 and WLS_IMG2 listen addresses to the list of addresses that are allowed to connect to Oracle WebCenter Content:

   ```
   SocketHostNameSecurityFilter=localhost|localhost.mycompany.com|WCCHOST1|
   WCCHOST2|WCCHOST1VHN1|WCCHOST2VHN1
   AlwaysReverseLookupForHost=Yes
   ```

4. Save the modified `config.cfg` file and restart the Oracle WebCenter Content servers for the changes to take effect.

11.17 Creating a Connection to Oracle WebCenter Content Server

To create a connection to Oracle WebCenter Content Server:

1. Log in to the WLS_IMG1 Imaging console at `http://WCCHOST1VHN1:16000/imaging`.
2. On the left, click **Manage Connections**, and then **Create Content Server Connection**.
3. Enter a name and description for the new connection, and then click **Next**.
4. In the Connection Settings screen, do the following:
   - Make sure the **Use Local Content Server** checkbox is selected.
   - Set the Content Server port to 4444.
   - Add two servers to the Content Server pool:
     - WCCHOST1:4444
     - WCCHOST2:4444

   Click **Next**.
5. In the Connection Security screen, leave the default selections for the WebLogic Server user, and then click **Next**.
6. Review the connection details and click **Submit**.

11.18 Configuring BPEL CSF Credentials

When connecting to a BPEL system from Oracle WebCenter Content: Imaging, you need to configure the required credential to communicate with Oracle SOA Suite. To add this credential, use these steps:

1. On SOAHOST1, change directory to the `common/bin` location under the Oracle WebCenter Content home in SOAHOST1 (where your Administration Server resides):

   ```
   cd WCC_ORACLE_HOME/common/bin
   ```

   (*WCC_ORACLE_HOME* is the Oracle home for Oracle WebCenter Content, which is *MW_HOME*`/wcc` in the EDG topology.)

2. Run the Oracle WebLogic Scripting Tool (WLST):

   ```
   ./wlst.sh
   ```

3. Run `connect()` and supply the user name, password, and Administration Server URL (`t3://ADMINVHN:7001`).

   ```
   wls:/offline> connect()
   ```

4. Create a CSF (Credential Store Framework) credential. This credential is the credential that Imaging will use to connect to the BPEL system. It should be a BPEL admin user. CSF credentials are user name/password pairs that are keyed by an *alias* and stored inside a named *map* in the CSF. Because of its integration with OWSM web services, Imaging is currently leveraging the standard OWSM CSF map named `oracle.wsm.security`. To create a credential, use the `createCred` WLST command:

   ```
   wls:/domain_name/serverConfig> createCred(map="oracle.wsm.security",
   key="basic.credential", user="weblogic", password="password_for_credential")
   ```

   The `key` value in the command is the *alias*, which is used for the `Credential Alias` property of the BPEL connection definition in the Imaging administration user interface (also the `Connection.CONNECTION_BPEL_CSFKEY_KEY` property in the API). The alias `basic.credential` is used in the example because it is a standard default name used by OWSM and BPEL. However, the alias can be anything as long as it is unique within the map.

   > **Note:** A new map will need to be created or the existing one updated if a different user and/or password is later used when integrating the Oracle SOA Suite system with a central LDAP and single sign-on (SSO) system. For details on the sample users created, see Chapter 15, "Integrating with Oracle Identity Management."

5. Restart the Oracle SOA Suite and Imaging Managed Servers.

11.19 Configuring a Workflow Connection

To configure a workflow connection:

1. Log in to the WLS_IMG1 imaging console at `http://WCCHOST1VHN1:16000/imaging`.
2. From the navigator pane, under Manage Connections, click the Add icon and then **Create Workflow Connection**.

3. On the Workflow Connection Basic Information page, enter a name for the connection. The name will display in the Manage Connections panel. This field is required. Optionally, enter a brief description of the connection. The connection type defaults to Workflow Connection.
4. Click **Next**.
5. In the Workflow Connection Settings Page, do the following:
   a. In the **HTTP Front End Address** field, specify the host name or IP address, domain, and port number of the workflow server: `http://soainternal.mycompany.com:80`. This field is required.
   b. In the **Credential Alias** field, provide the credential alias created earlier as described in Section 11.18, "Configuring BPEL CSF Credentials."
   c. In the **Provider** field, enter your two Oracle SOA Suite server listen addresses separated by a comma: `t3://SOAHOST1VHN1,SOAHOST2VHN1:8001`
   d. Click the **Test Connection** button to confirm the connection parameters and see what composites exist on that BPEL machine.
   e. Click **Next**.
6. Modify the security grants if desired.
7. Click **Next**.
8. Click **Submit**.

11.20 Configuring Oracle HTTP Server for the WLS_IMG Managed Servers

To enable Oracle HTTP Server to route to IMG_Cluster, which contains the WLS_IMG1 and WLS_IMG2 Managed Servers, you must set the `WebLogicCluster` parameter to the list of nodes in the cluster as follows:

1. For each of the web servers on WEBHOST1 and WEBHOST2, add the following lines to the *ORACLE_INSTANCE*`/config/OHS/ohs1/moduleconf/wcc_vh.conf` and *ORACLE_INSTANCE*`/config/OHS/ohs2/moduleconf/wcc_vh.conf` files:

```
# Oracle WebCenter Content: Imaging Application
<Location /imaging >
    WebLogicCluster WCCHOST1VHN1:16000,WCCHOST2VHN1:16000
    SetHandler weblogic-handler
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>

# AXF WS Invocation
<Location /axf-ws >
    WebLogicCluster WCCHOST1VHN1:16000,WCCHOST2VHN1:16000
    SetHandler weblogic-handler
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>
```

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

   ORACLE_BASE/admin/*instance_name*/bin/opmnctl restartproc ias-component=ohsX

   For WEBHOST1, use `ohs1` for `ias-component` and for WEBHOST2 use `ohs2`.

11.21 Setting the Front-End HTTP Host and Port

You must set the front-end HTTP host and port for the Imaging cluster (IMG_Cluster):

1. Log in to the WebLogic Server Administration Console.
2. Go to the Change Center section and click **Lock & Edit**.
3. Expand the **Environment** node in the **Domain Structure** tree on the left.
4. Click **Clusters**.
5. On the Summary of Clusters page, select **IMG_Cluster**.
6. Open the **HTTP** tab.
7. Set the following values:
   - **Frontend Host:** wcc.mycompany.com
   - **Frontend HTTPS Port:** 443
   - **Frontend HTTP Port:** 80
8. Click **Save**.
9. Click **Activate Changes** in the Change Center section of the Administration Console.
10. Restart the servers to make the front-end host directive in the cluster take effect.

11.22 Validating Access Through the Load Balancer

Verify URLs to ensure that appropriate routing and failover is working from the HTTP Server to the IMG_Cluster. To verify the URLs:

1. While WLS_IMG2 is running, stop WLS_IMG1 from the WebLogic Server Administration Console.
2. Access `http://wcc.mycompany.com/imaging` to verify it is functioning properly. (Please note that you will not be able to retrieve reports or data since the Imaging server is down.)
3. Start WLS_IMG1 from the WebLogic Server Administration Console.
4. Stop WLS_IMG2 from the WebLogic Server Administration Console.
5. Access `http://wcc.mycompany.com/imaging` to verify it is functioning properly.
6. Start WLS_IMG2 from the WebLogic Server Administration Console.

11.23 Configuring Node Manager for the WLS_IMG Managed Servers

It is assumed that the host names used by the WLS_IMG Managed Servers as listen addresses have already been configured for host name verification as explained in Section 10.14, "Configuring Node Manager for the WLS_WCC and WLS_IMG Managed Servers."

At this point, once the Imaging Managed Servers have been added to the domain, the procedure in Section 13.3.5, "Configuring Managed Servers to Use the Custom Keystores," should be performed so that the servers are configured to use custom key stores.

11.24 Configuring Server Migration for the WLS_IMG Managed Servers

Server migration is required for proper failover of the Oracle WebCenter Content: Imaging components in the event of failure in any of the WCCHOST1 and WCCHOST2 nodes. For more details, see Chapter 14, "Configuring Server Migration for an Enterprise Deployment." For Imaging, use the following values for the variables in that chapter:

- Server names:
  - *WLS_SERVER1*: WLS_IMG1
  - *WLS_SERVER2*: WLS_IMG2
- Host names:
  - *HOST1*: WCCHOST1
  - *HOST2*: WCCHOST2
- Cluster name:
  - *CLUSTER*: IMG_Cluster

11.25 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 that 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 that guide. For information on how to recover components, see the "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" section in the guide. For information about database backup, see the *Oracle Database Backup and Recovery User's Guide*.

To back up the installation at this point:

1. Back up the web tier on WEBHOST1:

   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 Oracle instance on the web tier using the following command:

   ```
   tar -cvpf BACKUP_LOCATION/web_instance_name.tar ORACLE_INSTANCE
   ```

   d. Start the instance using `opmnctl`:

   ```
   cd 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 operating system tools such as tar for cold backups if possible.

3. Back up the Administration Server and Managed Server domain directory to save your domain configuration. The configuration files all exist in the *ORACLE_BASE*`/admin/`*domain_name/* directory. Run the following command in SOAHOST1 to create the backup:

   ```
   tar -cvpf edgdomainback.tar ORACLE_BASE/admin/domain_name
   ```

12

Extending the Domain to Include Inbound Refinery

This chapter describes how to extend a domain to include Oracle WebCenter Content: Inbound Refinery using the Oracle Fusion Middleware Configuration Wizard. It contains the following sections:

- Section 12.1, "Overview of Extending the Domain to Include Oracle WebCenter Content: Inbound Refinery"
- Section 12.2, "Extending the Domain for Inbound Refinery"
- Section 12.4, "Restarting the Administration Server"
- Section 12.3, "Propagating the Domain Configuration to WCCHOST1 and WCCHOST2 Using the pack and unpack Utilities"
- Section 12.5, "Starting the Inbound Refinery Managed Servers"
- Section 12.6, "Configuring the Inbound Refineries"
- Section 12.7, "Validating the Configuration of the Inbound Refinery Managed Servers"

> **Note:** Before starting the setup process, read the *Oracle Fusion Middleware Release Notes* for your platform for additional installation and deployment information.

12.1 Overview of Extending the Domain to Include Oracle WebCenter Content: Inbound Refinery

Oracle WebCenter Content: Inbound Refinery is required for document conversion by Oracle WebCenter Content Server. The actual number of Inbound Refinery Managed Servers varies depending on requirements. For availability reasons, Oracle recommends configuring at least two Inbound Refinery Managed Servers. Configure these servers on separate machines. In the reference Oracle WebCenter Content enterprise deployment topology, Inbound Refinery will be configured on the same machine as Content Server.

Even though a cluster is created in the process of extending the domain with Inbound Refinery, each Inbound Refinery instance is completely independent. Clustering is used for management purposes only.

Extend the domain to include Oracle WebCenter Content: Inbound Refinery. Table 12–1 lists the steps for configuring WebCenter Content and other tasks required for extending the domain with Inbound Refinery Managed Servers.

Table 12–1 Steps for Extending the Domain with Inbound Refinery

Step	Description	More Information
Extend the domain for Inbound Refinery	Extend the Oracle WebLogic Server domain you created in Chapter 8, "Creating a Domain for an Enterprise Deployment."	Section 12.2, "Extending the Domain for Inbound Refinery"
Propagate the domain configuration to the Inbound Refinery Managed Servers	Propagate the start scripts and classpath configuration from the Administration Server's domain directory to the Managed Server domain directories.	Section 12.3, "Propagating the Domain Configuration to WCCHOST1 and WCCHOST2 Using the pack and unpack Utilities"
Restart the Administration Server for the domain	Stop and then start the Administration Server to make the changes from the previous step take effect.	Section 12.4, "Restarting the Administration Server"
Start the Inbound Refinery Managed Servers	Start the WLS_IBR1 and WLS_IBR2 Managed Servers.	Section 12.5, "Starting the Inbound Refinery Managed Servers"
Configure the Inbound Refinery instances	Complete the initial configuration of Inbound Refinery on WLS_IBR1 and WLS_IBR	Section 12.6, "Configuring the Inbound Refineries"
Verify the Inbound Refinery configuration	Verify that a file with an extension recognized as valid for conversion is correctly converted on Content Server.	Section 12.7, "Validating the Configuration of the Inbound Refinery Managed Servers"

12.2 Extending the Domain for Inbound Refinery

To complete the enterprise deployment topology, you must extend the domain created in Chapter 8, "Creating a Domain for an Enterprise Deployment," to include Oracle WebCenter Content: Inbound Refinery.

> **Note:** Before performing these steps, back up the domain as described in the *Oracle Fusion Middleware Administrator's Guide*.

To extend the domain for Inbound Refinery:

1. Ensure that the database where you created the repository schema is running. For Oracle RAC databases, it is recommended that all instances are running, so that the validation check later on becomes more reliable.
2. On SOAHOST1, change the directory to the location of the Oracle Fusion Middleware Configuration Wizard. This is within the Oracle Common home directory (notice that domain extensions are run from the node where the Administration Server resides).

   ```
   cd ORACLE_COMMON_HOME/common/bin
   ```

3. Start the Configuration Wizard:

   ```
   ./config.sh
   ```

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

5. In the Select a WebLogic Domain Directory screen, select the WebLogic Server domain directory (*ORACLE_BASE*/admin/*domain_name*/aserver/*domain_name*), and click **Next**.
6. In the Select Extension Source screen, do the following:
   - Select **Extend my domain automatically to support the following added products**.
   - Select the following product:

     Oracle Universal Content Management - Inbound Refinery - 11.1.1.0 [wcc]

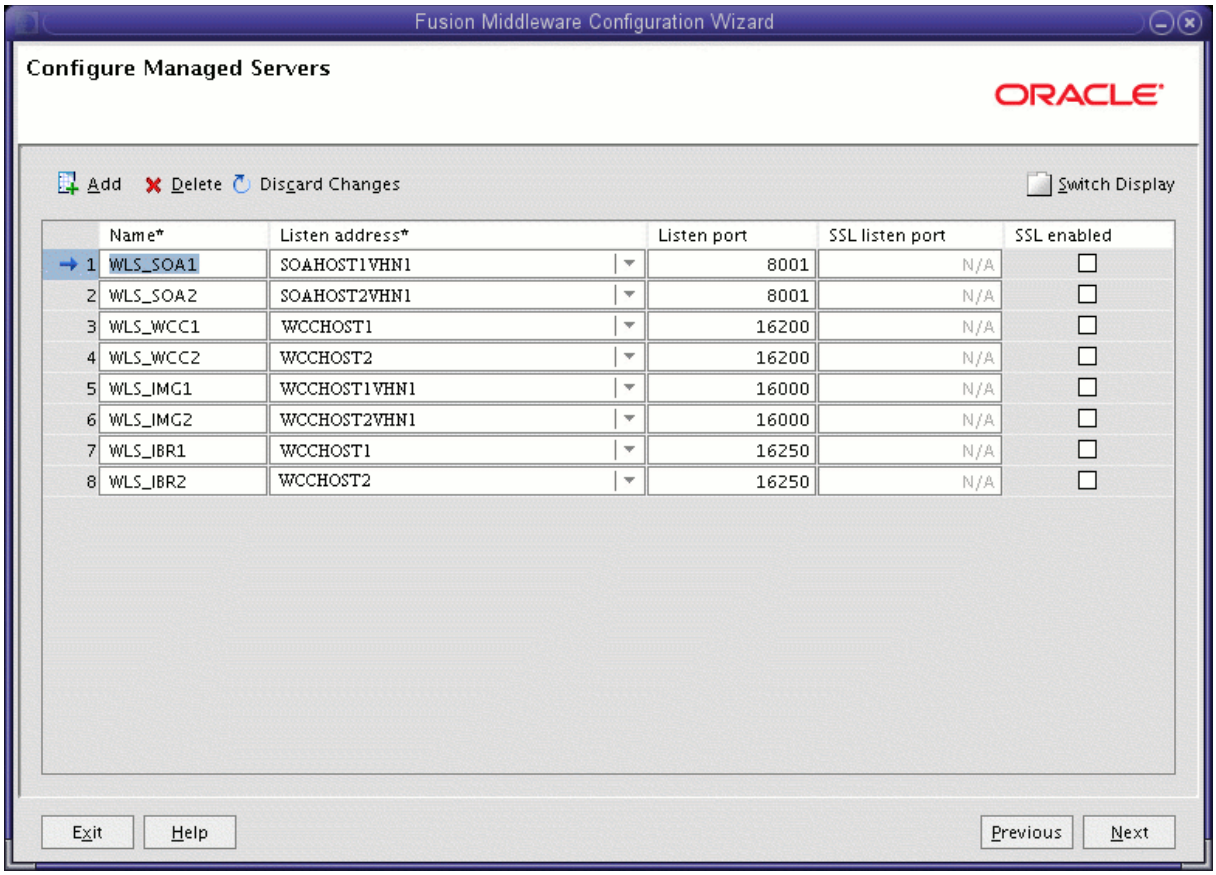
     This is the selection for Oracle WebCenter Content: Inbound Refinery.

   The following products should already be selected and grayed out. They were selected when you created a domain in Section 8.3, "Running the Configuration Wizard on SOAHOST1 to Create a Domain," or extended the domain in Section 9.3, "Extending the Domain for Oracle SOA Suite Components," Section 10.2, "Extending the Domain for WebCenter Content," or Section 11.3, "Extending the Domain for Imaging":
   - **Basic WebLogic Server Domain**
   - **Oracle SOA Suite for developers**
   - **Oracle SOA Suite**
   - **Oracle WebCenter Content: Imaging**
   - **Oracle Universal Content Management - Content Server**
   - **Oracle Enterprise Manager**
   - **Oracle WSM Policy Manager**
   - **Oracle JRF**

   Click **Next**.
7. In the Configure GridLink RAC Component Schema screen, nothing needs to be done. Inbound Refinery does not have a schema in the database. Click **Next** to continue.
8. In the Test JDBC Component Schema screen, nothing needs to be done. Click **Next** to continue.
9. In the Optional Configuration screen, select the following options:
   - **Managed Servers, Clusters and Machines**
   - **Deployment and Services**

   Click **Next**.
10. In the Configure Managed Servers screen (Figure 12–1), add a Managed Server for Inbound Refinery and configure the Inbound Refinery servers.

Figure 12–1 Configure Inbound Refinery Managed Servers



One server is created automatically. Rename this server to WLS_IBR1, and add a new server named WLS_IBR2. Give these servers the attributes listed in Table 12–2. Do not modify the other servers that are shown in this screen; leave them as they are.

Table 12–2 Managed Servers

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled
WLS_IBR1	WCCHOST1	16250	n/a	No
WLS_IBR2	WCCHOST2	16250	n/a	No

Click **Next**.

11. In the Configure Clusters screen, click **Add** to add the Inbound Refinery cluster shown in Table 12–3. Do not modify the other clusters that appear in this screen; leave them as they are.

Table 12–3 Cluster Configuration

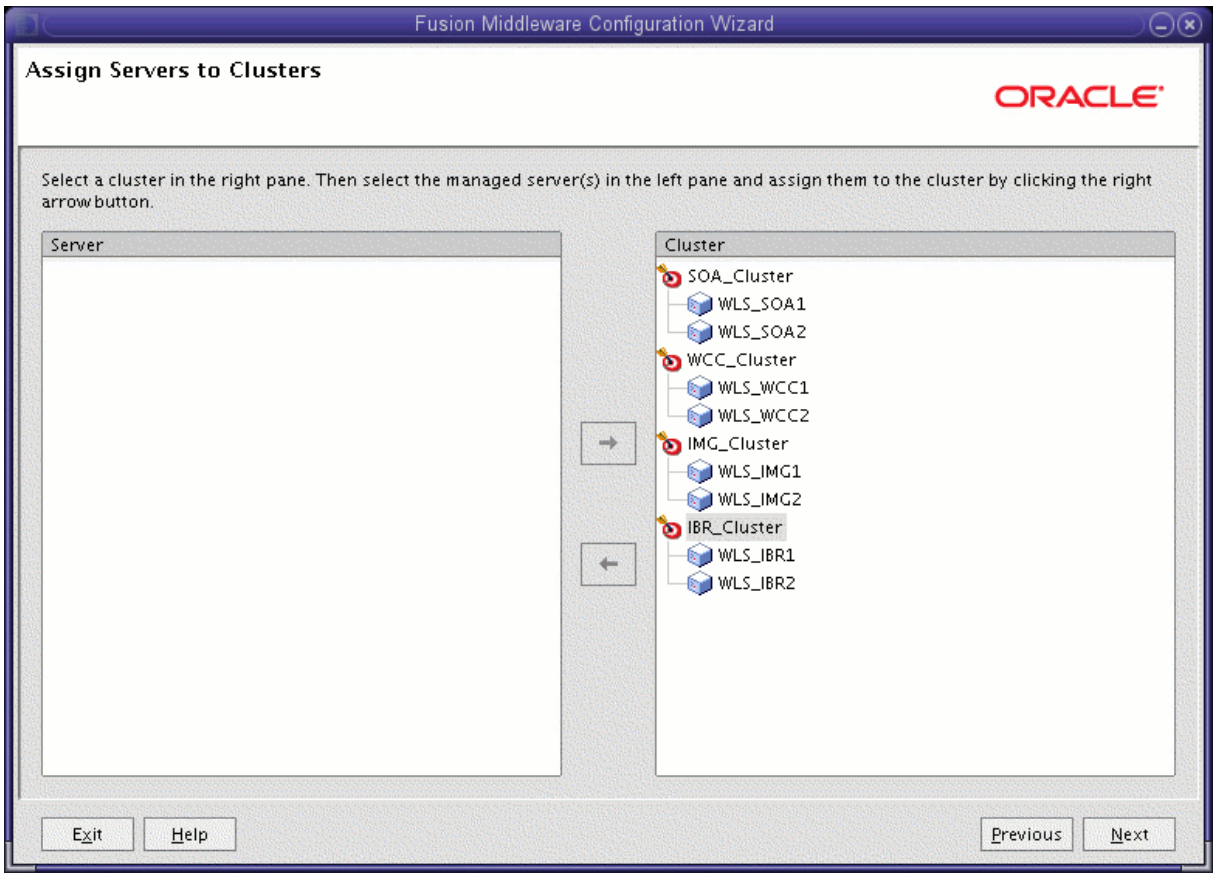
Name	Cluster Messaging Mode	Multicast Address	Multicast Port	Cluster Address
IBR_Cluster	unicast	n/a	n/a	Leave empty

Click **Next**.

Note: All Inbound Refinery instances are completely independent. The cluster is used for management purposes only.

12. In the Assign Servers to Clusters screen (Figure 12–2), add the following items:
    - `IBR_Cluster:`
      - `WLS_IBR1`
      - `WLS_IBR2`

Figure 12–2 Assign Inbound Refinery Servers to a Cluster

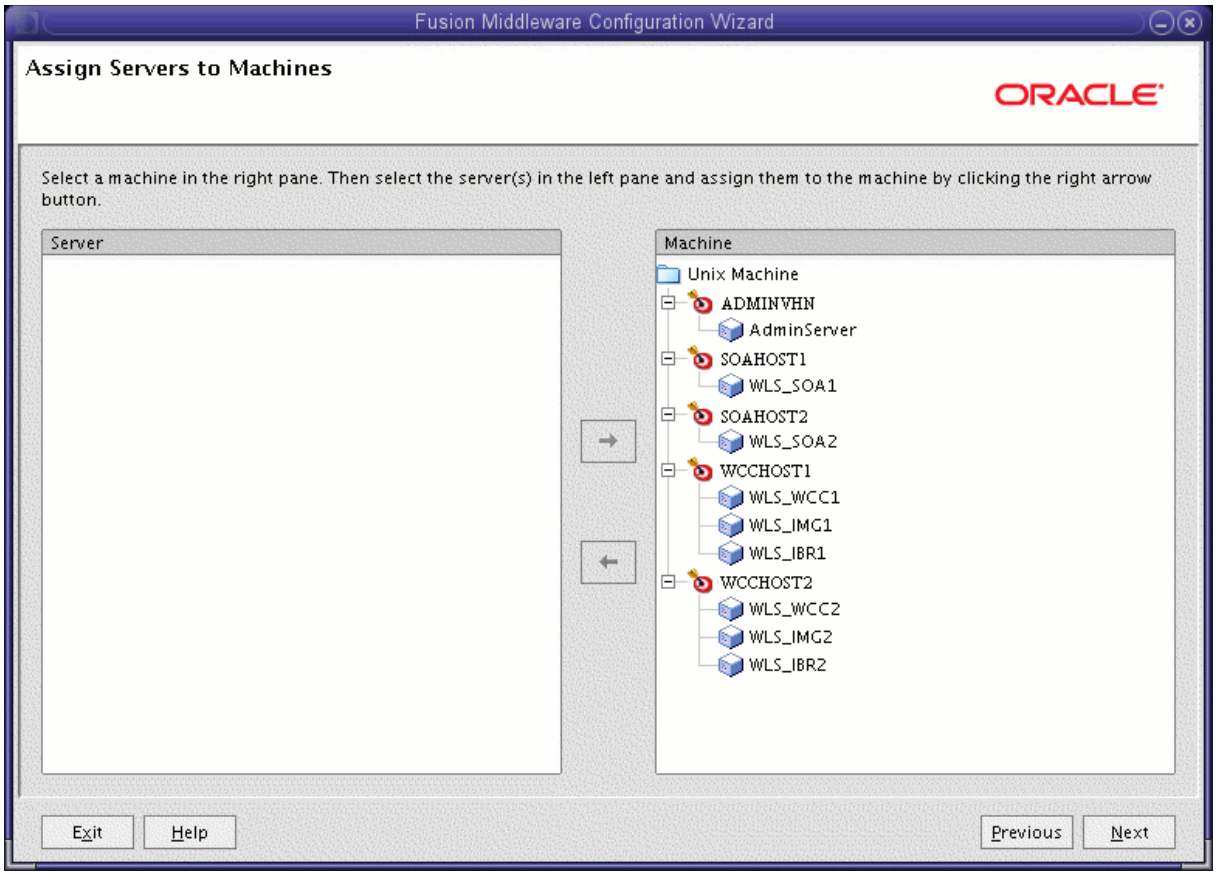


Do not modify the other assignments that appear in this screen; leave them as they are.

Click **Next**.

13. In the Configure Machines screen, click **Next**.
14. In the Assign Servers to Machines screen (Figure 12–3), assign the Inbound Refinery Managed Servers to machines as follows:
    - Assign **WLS_IBR1** to **WCCHOST1**.
    - Assign **WLS_IBR2** to **WCCHOST2**.

Figure 12–3 Assign Inbound Refinery Servers to Machines



Click **Next**.

15. In the Target Deployments to Clusters or Servers screen, verify the following targets:

    - The **usermessagingserver** and **usermessagingdriver-email** deployments should be targeted only to **SOA_Cluster**. (The **usermessaging-xmpp**, **usermessaging-smpp**, and **usermessaging-voicexml** applications are optional.)
    - **WSM-PM** should be targeted only to **SOA_Cluster**.
    - The **oracle.rules***, **oracle.sdp.*** and **oracle.soa.*** deployments should be targeted only to **SOA_Cluster**, except for the **oracle.soa.workflow.wc** library, which should be targeted to both **SOA_Cluster** and **IMG_Cluster**.

    Click **Next**.

16. In the Target Services to Clusters or Servers screen, click **Next**.
17. In the Configuration Summary screen, click **Extend**.
18. In the Creating Domain screen, click **Done**.
19. Restart the Administration Server to make these changes take effect.

    To restart the Administration Server, stop it first using the Administration Console and then start it again as described in Section 8.4.3, "Starting the Administration Server on SOAHOST1."

12.3 Propagating the Domain Configuration to WCCHOST1 and WCCHOST2 Using the pack and unpack Utilities

To propagate the domain configuration:

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

```
cd ORACLE_COMMON_HOME/common/bin

./pack.sh -managed=true -domain=ORACLE_BASE/admin/domain_name/aserver/domain_
name -template=edgdomaintemplateIBR.jar -template_name=edgdomain_templateIBR
```

2. Run the following command on SOAHOST1 to copy the template pack created in the previous step to WCCHOST2:

> **Note:** Assuming that WCCHOST1 shares the Oracle home with SOAHOST1, the template will be present in the same directory in WCCHOST1; otherwise, copy it also to WCCHOST1.

```
scp edgdomaintemplateIBR.jar oracle@WCCHOST2:ORACLE_BASE/product/fmw/oracle_
common/common/bin
```

3. Run the `unpack` command on WCCHOST1 to unpack the propagated template.

> **Note:** Make sure to run `unpack` from the *ORACLE_COMMON_HOME*`/common/bin/` directory, not from *WL_HOME*`/common/bin/`.

```
cd ORACLE_COMMON_HOME/common/bin

./unpack.sh -domain=ORACLE_BASE/admin/domain_name/mserver/domain_name
-template=edgdomaintemplateIBR.jar -app_dir=ORACLE_BASE/admin/domain_
name/mserver/applications -overwrite_domain=true
```

> **Note:** The *ORACLE_BASE*`/admin/`*domain_name*`/mserver/` directory must exist before you run `unpack`. In addition, the *ORACLE_BASE*`/admin/`*domain_name*`/mserver/applications/` directory must be empty.

4. Repeat step 3 for WCCHOST2.

12.4 Restarting the Administration Server

Restart the Administration Server to make these changes take effect. To restart the Administration Server, stop it first using the Administration Console and then start it again as described in Section 8.4.3, "Starting the Administration Server on SOAHOST1."

12.5 Starting the Inbound Refinery Managed Servers

To start the WLS_IBR1 Managed Server on WCCHOST1:

1. Log in to the Oracle WebLogic Server Administration Console at `http://ADMINVHN:7001/console`.
2. In the **Domain Structure** tree on the left, expand the **Environment** node, and then select **Servers**.
3. On the Summary of Servers page, open the **Control** tab.
4. Select WLS_IBR1 from the **Servers** column of the table.
5. Click **Start**.

Repeat these steps to start the WLS_IBR2 Managed Server on WCCHOST2.

12.6 Configuring the Inbound Refineries

An inbound refinery needs to be accessed only once through HTTP in order to initialize its configuration. This can be done directly, at the Managed Server's listen address. An inbound refinery should not be placed behind an HTTP server.

All subsequent access to an inbound refinery is through the socket listener. This listener is protected through the incoming socket connection address security filter configured in the next section.

Oracle recommends configuring each Oracle WebCenter Content Server with all Inbound Refinery instances. The process for configuring Oracle WebCenter Content is to add each Inbound Refinery instance as a provider. There are also post-installation steps that must be performed with Inbound Refinery.

The following sections describe the procedures for post-installation configuration of each Inbound Refinery instance:

- Section 12.6.1, "Configuring Inbound Refinery Settings"
- Section 12.6.2, "Specifying the Font Path"
- Section 12.6.3, "Configuring Document Conversion"
- Section 12.6.4, "Setting Up Content Server to Send Jobs to Inbound Refinery for Conversion"

12.6.1 Configuring Inbound Refinery Settings

To configure the settings for each Inbound Refinery instance:

1. Access the Inbound Refinery post-installation configuration screen at the following URL, in which *N* is `1` or `2`:

   `http://WCCHOSTN:16250/ibr/`

2. In the Configuration screen, you will see Inbound Refinery Instance Identifier: *name*. Set the configuration settings for this instance as follows:
   - **Inbound Refinery Instance Folder:** Set this to *ORACLE_BASE*`/admin/`*domain_name*`/`*ibr_cluster_name*`/ibr`*N*. The directory path should be on a shared disk, but should be unique for each Inbound Refinery instance.
   - **Native File Repository Location:** Set this to *ORACLE_BASE*`/admin/`*domain_name*`/`*ibr_cluster_name*`/ibr`*N*`/vault`.

- **WebLayout Folder:** Set this to *ORACLE_BASE*/admin/*domain_name*/*ibr_cluster_name*/ibr*N*/weblayout.
- **User Profile Folders:** Set this to *ORACLE_BASE*/admin/*domain_name*/*ibr_cluster_name*/ibr*N*/data/users/profiles.
- **Incoming Socket Connection Address Security Filter:** Set this to a pipe-delimited list of localhost and the server IPs:

  127.0.0.1|*WCCHOST1-IP*|*WCCHOST2-IP*|*WEBHOST1-IP*|*WEBHOST2-IP*

  This enables access from Oracle WebCenter Content Server. The values for *WCCHOST1-IP* and *WCCHOST2-IP* should be the IP addresses of the machines with the Oracle WebCenter Content Server instance or instances that will send jobs to Inbound Refinery, not necessarily the IP address of Inbound Refinery. (In the reference topology used in this enterprise deployment guide, however, these IP addresses are the same.)

  This field accepts wildcards in the value; for example, 192.0.2.*. You can change this value later by setting SocketHostAddressSecurityFilter in *ORACLE_BASE*/admin/*domain_name*/mserver/*domain_name*/ucm/ibr/config/config.cfg and restarting Inbound Refinery.
- **Server Socket Port:** Enter an unused port number, such as 5555. This value is the number of the port for calling top-level services. Changing this field value changes the IntradocServerPort entry in *ORACLE_BASE*/admin/*domain_name*/mserver/*domain_name*/ucm/ibr/config/config.cfg. Take note of the port number as you need it later when configuring Oracle WebCenter Content.
- **Server Instance Name:** Specify a name for the Inbound Refinery server instance. You can accept the default or change it to a more useful name if you want. Take note of the server name as you need it later when configuring Oracle WebCenter Content.

You can leave all other fields on the configuration page as they are.

Click **Submit**, and you should got the following message:

```
Post-install configuration complete. Please restart this node.
```

3. Restart the Inbound Refinery Managed Server.
4. Repeat these steps for all the inbound refineries, using different names for the content folders.

12.6.2 Specifying the Font Path

For Inbound Refinery to work properly, you must specify the path to fonts used to generate font images. By default, the font path is set to the font directory in the JVM used by Inbound Refinery: *MW_HOME*/jdk160_*version*/jre/lib/fonts. However, the fonts included in the default directory are limited and may cause poor renditions. Also, in some cases if a non-standard JVM is used, then the JVM font path may be different than that specified as the default. If this is the case, an error message is displayed from both Inbound Refinery and Content Server. If this occurs, ensure the font path is set to the directory containing the fonts necessary to properly render your conversions. For more information, see "Specifying the Font Path" in *Oracle Fusion Middleware Managing Oracle WebCenter Content*.

12.6.3 Configuring Document Conversion

To configure document conversion on each Inbound Refinery Managed Server:

1. Log in to Inbound Refinery at the following URL, in which `N` is `1` or `2`

   `http://WCCHOSTN:16250/ibr/`

2. Enable conversion components on Inbound Refinery. The core Inbound Refinery converts files to TIFF web-viewable files and JPEG image thumbnails. To use additional conversion types, you need to enable the necessary components:
   a. Open the **Administration** tray or menu, then choose **Admin Server**.
   b. In the Component Manager, select **PDFExportConverter** in Inbound Refinery and any other components you want. PDFExportConverter uses Outside In to convert documents directly to PDF files. The conversion can be cross-platform and does not require any third-party product. You can enable PDFExportConverter for Inbound Refinery as a server feature. For more information, consult the readme files and the documentation for each component.
   c. Click **Update**.
   d. Click **OK** to enable the components.
   e. Restart the Inbound Refinery Managed Server.

   Note: For information about the conversion components, see "Configuring the Content Server Refinery Conversion Options" in *Oracle Fusion Middleware Managing Oracle WebCenter Content*.

3. Restart the Administration Server and all Inbound Refinery Managed Servers.
4. Set the primary web-viewable conversion to PDF Export:
   a. Select **Conversion Settings**, then select **Primary Web Rendition**.
   b. On the Primary Web-Viewable Rendition page, select **Convert to PDF using PDF Export**.
   c. Click **Update** to save your changes.

   Inbound Refinery will now use Outside In PDF Export to convert files directly to PDF without the use of third-party applications.

12.6.4 Setting Up Content Server to Send Jobs to Inbound Refinery for Conversion

Before Oracle WebCenter Content Server can send jobs to Inbound Refinery for conversion, you need to perform these setup tasks for each Inbound Refinery Managed Server:

- Section 12.6.4.1, "Creating an Outgoing Provider"
- Section 12.6.4.2, "Enabling Components for Inbound Refinery on Content Server"
- Section 12.6.4.3, "Selecting File Formats To Be Converted"

12.6.4.1 Creating an Outgoing Provider

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

To create an outgoing provider for each Inbound Refinery:

1. Log in to Oracle WebCenter Content Server at the following URL:

   `http://WCCHOST1:16200/cs/`

2. Open the **Administration** tray or menu and choose **Providers**.
3. In the Create a New Provider section of the Providers page, click **Add** in the **outgoing** row.
4. Enter the following values for the fields:
   - **Provider Name:** Any short name with no spaces. It is a good idea to use the same value as the **Instance Name** value
   - **Provider Description:** Any text string.
   - **Server Host Name:** The name of the host machine where the Inbound Refinery instance is running: `WCCHOST1`.
   - **HTTP Server Address:** The address of the Inbound Refinery instance: `WCCHOST1:16250`.
   - **Server Port:** The value of the Server Socket Port field for the Inbound Refinery instance as specified in Section 12.6.1, "Configuring Inbound Refinery Settings"; for example, `5555`. This is the `IntradocServerPort` value in the Content Server `config.cfg` file.
   - **Instance Name:** The server instance name for Inbound Refinery as specified in Section 12.6.1, "Configuring Inbound Refinery Settings." This is the `IDC_Name` value in the Content Server's config.cfg file.
   - **Relative Web Root**: The web root of the Inbound Refinery instance: `/ibr/`.
5. Under Conversion Options, check **Handles Inbound Refinery Conversion Jobs.**

   Do *not* check **Inbound Refinery Read Only Mode**.
6. Click **Add**.
7. Restart the Inbound Refinery Managed Server and Oracle WebCenter Content Server.
8. Go back to the Providers page, and check that the **Connection State** value is `good` for the provider.

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

For more information about setting up providers, see "Configuring Content Server and Refinery Communication" in *Oracle Fusion Middleware Managing Oracle WebCenter Content*.

12.6.4.2 Enabling Components for Inbound Refinery on Content Server

Some conversion types require *helper* components to be enabled on Content Server. The `InboundRefinerySupport` component must always be enabled on any Content Server instance that uses Inbound Refinery for document conversion. It is enabled by default on a new Content Server installation.

To enable Inbound Refinery components on Content Server:

1. Log in to Oracle WebCenter Content Server at the following URL:

   `http://WCCHOST1:16200/cs/`

2. Open the **Administration** tray or menu, then choose **Admin Server**, and then **Component Manager**.
3. Under Inbound Refinery, make sure **InboundRefinerySupport** is selected, and select any other components, such as **XMLConverterSupport**, that you want to enable.
4. Click **Update**.
5. Restart Oracle WebCenter Content Server.

12.6.4.3 Selecting File Formats To Be Converted

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

To select file formats to be converted:

1. Log in to Oracle WebCenter Content Server at the following URL:

   `http://WCCHOST1:16200/cs/`

2. Open the **Administration** tray or menu, then choose **Refinery Administration**, and then **File Formats Wizard** to open the File Formats Wizard page, which specifies what file formats will be sent to Inbound Refinery for conversion when they are checked into Content Server.
3. Select the formats you want converted, such as **doc, dot**, **docx**, and **dotx** for Microsoft Word documents.
4. Click **Update**.

You can also select file formats with the Configuration Manager, with more fine-grained control, including file formats that wizard does not list. For more information, see "Managing File Types" in *Oracle Fusion Middleware Managing Oracle WebCenter Content*.

12.7 Validating the Configuration of the Inbound Refinery Managed Servers

To ensure that the Inbound Refinery Managed Servers you have created are properly configured, validate the configuration by logging in to Oracle WebCenter Content Server and verifying that a file with an extension recognized as valid for conversion is correctly converted.

For example, if you selected **docx** as a format to be converted, you can convert a Microsoft Word document with a `.docx` extension to PDF format.

For information about the check-in and check-out procedures, see "Uploading Documents" and "Checking Out and Downloading Files" in *Oracle Fusion Middleware Using Oracle WebCenter Content*.

For information about the conversion process, see "Configuring Content Servers to Send Jobs to Refineries" in *Oracle Fusion Middleware Managing Oracle WebCenter Content*.

13

Setting Up Node Manager

This chapter describes how to configure Node Manager in accordance with the EDG recommendations. It contains the following sections:

- Section 13.1, "Overview of Setting Up Node Manager"
- Section 13.2, "Changing the Location of the Node Manager Log"
- Section 13.3, "Enabling Host Name Verification Certificates for Node Manager"
- Section 13.4, "Starting Node Manager"

13.1 Overview of Setting Up Node Manager

Node Manager enables you to start and stop the Administration Server and the Managed Servers for an Oracle WebLogic Server domain. Table 13–1 describes the setup steps, followed by an overview of the setup process and Oracle recommendations for the location of the Node Manager log and host name verification.

Table 13–1 Steps for Setting Up Node Manager

Step	Description	More Information
Specify a location for the Node Manager log within the admin directory for the ED.	Edit the `LogFile` property in the nodemanager.properties file, and restart Node Manager.	Section 13.2, "Changing the Location of the Node Manager Log"
Set up SSL communication between Node Manager and the Administration Server	Generate self-signed certificates, create identity and trust keystores, configure Node Manager and the Managed Servers to use the custom keystores, and change the host name verification setting for each Managed Server.	Section 13.3, "Enabling Host Name Verification Certificates for Node Manager"
Start Node Manager	Run the setNMProps.sh script before starting Node Manager the first time.	Section 13.4, "Starting Node Manager"

Process

The procedures described in this chapter must be performed for various components of the enterprise deployment topology outlined in Section 2.1.1, "Reference Topology Documented in the Guide." Variables are used in this chapter to distinguish between component-specific items:

- *WLS_SERVER*: this refers to a WebLogic Server Managed Server for the enterprise deployment component (for example, WLS_SOA1).
- *HOST*: this refers to a host machine for the enterprise deployment component (for example, SOAHOST1).
- *VIP*: this refers to a virtual IP for the enterprise deployment component (for example, SOAHOST1VHN1).

The values to be used to these variables are provided in the component-specific chapters in this EDG. Please note that the procedures in this chapter must be performed multiple times for each VIP-and-IP pair using the information provided in the component-specific chapters.

Recommendations

Oracle provides two main recommendations for Node Manager configuration in enterprise deployment topologies:

1. Oracle recommends placing the Node Manager log file in a location different from the default one (which is inside the Middleware home where Node Manager resides). See Section 13.2, "Changing the Location of the Node Manager Log" for further details.
2. Oracle also recommends using host name verification for the communication between Node Manager and the servers in the domain. This requires the use of certificates for the different addresses used in the domain. This chapter explains the steps for configuring certificates in the hosts for host name verification. See Section 13.3, "Enabling Host Name Verification Certificates for Node Manager" for further details.

Note: The passwords used in this guide are used only as examples. Use secure passwords in a production environment. For example, use passwords that consist of random sequences of both uppercase and lowercase characters as well as numbers.

13.2 Changing the Location of the Node Manager Log

Edit the Node Manager properties file located at *MW_HOME*`/wlserver_10.3/common/nodemanager/nodemanager.properties`. Add the new location for the log file, using the following line:

```
LogFile=ORACLE_BASE/admin/nodemanager.log
```

Oracle recommends that this location is outside the *MW_HOME* directory and inside the admin directory for the EDG.

Restart Node Manager for the change to take effect.

13.3 Enabling Host Name Verification Certificates for Node Manager

Setting up SSL for communication between Node Manager and the Administration Server consists of the following steps:

- 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
- Step 5: Configuring Managed Servers to Use the Custom Keystores
- Step 6: Changing the Host Name Verification Setting for the Managed Servers

13.3.1 Generating Self-Signed Certificates Using the utils.CertGen Utility

The certificates added in this chapter (as an example) address a configuration where Node Manager listens on a physical host name (*HOST*.mycompany.com) and a WLS Managed Server listens on a virtual host name (*VIP*.mycompany.com). Whenever a server is using a virtual host name, it is implied that the server can be migrated from one node to another. Consequently, the directory where keystores and trust keystores are maintained ideally must reside on a shared storage that is accessible from the failover. If additional host names are used in the same or different nodes, the steps in this example will need to be extended to:

1. Add the required host names to the certificate stores (if they are different from *HOST*.mycompany.com and *VIP*.mycompany.com).
2. Change the identity and trust store location information for Node Manager (if the additional host names are used by Node Manager) or for the servers (if the additional host names are used by Managed Servers).

Follow the steps below to create self-signed certificates on *HOST*. These certificates should be created using the network name or alias. For information on using trust CA certificates instead, see "Configuring Identity and Trust" in *Oracle Fusion Middleware Securing Oracle WebLogic Server*. The examples below configure certificates for *HOST*.mycompany.com and *VIP*.mycompany.com; that is, it is assumed that both a physical host name (*HOST*) and a virtual host name (*VIP*) are used in *HOST*. It is also assumed that *HOST*.mycompany.com is the address used by Node Manager and *VIP*.mycompany.com is the address used by a Managed Server or the Administration Server. This is the common situation for nodes hosting an Administration Server and a Fusion Middleware component, or for nodes where two Managed Servers coexist with one server listening on the physical host name and one server using a virtual host name (which is the case for servers that use migration servers).

1. Set up your environment by running the *WL_HOME*`/server/bin/setWLSEnv.sh` script. In the Bourne shell, run the following commands on *HOST*:

   ```
   cd WL_HOME/server/bin

   . ./setWLSEnv.sh
   ```

   Verify that the `CLASSPATH` environment variable is set:

   ```
   echo $CLASSPATH
   ```

2. The directory where keystores and trust keystores are maintained must be on shared storage that is accessible from all nodes so that when the servers fail over (manually or with server migration), the appropriate certificates can be accessed from the failover node. Oracle recommends using central or shared stores for the certificates used for different purposes (like SSL set up for HTTP invocations, and so on). In this case, SOAHOST2 uses the cert directory created for SOAHOST1 certificates. Create a user-defined directory for the certificates:

   ```
   mkdir cert
   ```

3. Change directory to the directory that you just created:

   ```
   cd cert
   ```

4. Run the utils.CertGen tool from the user-defined directory on *HOST* to create the certificates for both *HOST*. mycompany.com and *VIP*. mycompany.com.

   Syntax (all on a single line):

   ```
   java utils.CertGen Key_Passphrase Cert_File_Name Key_File_Name
   [export | domestic] [Host_Name]
   ```

   Examples:

   ```
   java utils.CertGen password HOST.mycompany.com_cert HOST.mycompany.com_key
   domestic HOST.mycompany.com

   java utils.CertGen password VIP.mycompany.com_cert VIP.mycompany.com_key
   domestic VIP.mycompany.com
   ```

13.3.2 Creating an Identity Keystore Using the utils.ImportPrivateKey Utility

Follow these steps to create an identity keystore on *HOST*:

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*`/cert/`).

   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.

2. On *HOST*, import the certificate and private key for both *HOST*`.mycompany.com` and *VIP*`.mycompany.com` into the identity store. Make sure that you use a different alias for each of the certificate/key pairs imported.

   Syntax (all on a single line):

   ```
   java utils.ImportPrivateKey Keystore_File Keystore_Password
   Certificate_Alias_to_Use Private_Key_Passphrase
   Certificate_File
   Private_Key_File
   [Keystore_Type]
   ```

Examples:

```
java utils.ImportPrivateKey appIdentityKeyStore.jks password
appIdentity1 password
ORACLE_BASE/admin/domain_name/cert/HOST.mycompany.com_cert.pem
ORACLE_BASE/admin/domain_name/cert/HOST.mycompany.com_key.pem

java utils.ImportPrivateKey appIdentityKeyStore.jks password
appIdentity2 password
ORACLE_BASE/admin/domain_name/cert/VIP.mycompany.com_cert.pem
ORACLE_BASE/admin/domain_name/cert/VIP.mycompany.com_key.pem
```

13.3.3 Creating a Trust Keystore Using the Keytool Utility

Follow these steps to create the trust keystore on *HOST*:

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 keystore directly. Copy the standard Java keystore CA certificates on *HOST* 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/cert/
   appTrustKeyStore.jks
   ```

2. The default password for the standard Java keystore is changeit. Oracle recommends always changing the default password. Use the keytool utility on *HOST* to do this. The syntax is:

   ```
   keytool -storepasswd -new New_Password -keystore Trust_Keystore -storepass
   Original_Password
   ```

   For example:

   ```
   keytool -storepasswd -new password -keystore appTrustKeyStore.jks -storepass
   changeit
   ```

3. The CA certificate CertGenCA.der is used to sign all certificates generated by the utils.CertGen tool. It is located in the *WL_HOME*/server/lib/ directory. This CA certificate must be imported into appTrustKeyStore using the keytool utility on *HOST*. The syntax is:

   ```
   keytool -import -v -noprompt -trustcacerts -alias Alias_Name
   -file CA_File_Location -keystore Keystore_Location -storepass Keystore_Password
   ```

   For example:

   ```
   keytool -import -v -noprompt -trustcacerts -alias clientCACert -file
   WL_HOME/server/lib/CertGenCA.der -keystore appTrustKeyStore.jks -storepass
   password
   ```

13.3.4 Configuring Node Manager to Use the Custom Keystores

To configure 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_Password
CustomIdentityAlias=Identity_Keystore_Alias
CustomIdentityPrivateKeyPassPhrase=Private_Key_Used_When_Creating_Certificate
```

Make sure to use the correct value for `CustomIdentityAlias` on each node; that is, the custom identity alias specifically assigned to that node, for example for ...HOST2:

```
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityKeyStoreFileName=ORACLE_BASE/admin/domain_
name/cert/appIdentityKeyStore.jks
CustomIdentityKeyStorePassPhrase=password
CustomIdentityAlias=appIdentity2
CustomIdentityPrivateKeyPassPhrase=password
```

The passphrase entries in the nodemanager.properties file get encrypted when you start Node Manager as described in Section 13.4, "Starting Node Manager." 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 in Section 13.3.2, "Creating an Identity Keystore Using the utils.ImportPrivateKey Utility." 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 on *HOST* before starting Node Manager in the different nodes:

```
cd WL_HOME/server/bin

export JAVA_OPTIONS=-DCustomIdentityAlias=appIdentityX
```

Note: Make sure to specify the custom identity alias specifically assigned to each host, so `appIdentity1` for ...HOST1 and `appIdentity2` for ...HOST2.

13.3.5 Configuring Managed Servers to Use the Custom Keystores

Follow these steps to configure the identity and trust keystores for *WLS_SERVER*:

1. Log in to the Oracle WebLogic Server Administration Console.
2. Click **Lock & Edit**.
3. Expand the **Environment** node in the **Domain Structure** tree on the left.
4. Click **Servers**.
5. On the Summary of Server page, click the name of the server for which you want to configure the identity and trust keystores (*WLS_SERVER*).
6. On the settings page for the selected server, select **Configuration** and then **Keystores**.
7. Click the **Change** button next to the Keystores field and select the **Custom Identity and Custom Trust** method for storing and managing private keys/digital certificate pairs and trusted CA certificates. Click **Save** when you are done.

8. In the Identity section, define attributes for the identity keystore:
   - **Custom Identity Keystore:** The fully qualified path to the identity keystore:

     `ORACLE_BASE/admin/domain_name/cert/appIdentityKeyStore.jks`

   - **Custom Identity Keystore Type:** Leave blank; it defaults to JKS.
   - **Custom Identity Keystore Passphrase:** The password (*Keystore_Password*) you provided in Section 13.3.3, "Creating a Trust Keystore Using the Keytool Utility." This attribute is optional or required depending on the type of keystore. All keystores require the passphrase to write to the keystore. However, some keystores do not require the passphrase to read from the keystore. WebLogic Server reads only from the keystore, so whether or not you define this property depends on the requirements of the keystore.
9. In the Trust section, define properties for the trust keystore:
   - **Custom Trust Keystore:** The fully qualified path to the trust keystore:

     `ORACLE_BASE/admin/domain_name/cert/appTrustKeyStore.jks`

   - **Custom Trust Keystore Type:** Leave blank; it defaults to JKS.
   - **Custom Trust Keystore Passphrase:** The password you provided as *New_Password* in Section 13.3.3, "Creating a Trust Keystore Using the Keytool Utility." This attribute is optional or required depending on the type of keystore. All keystores require the passphrase to write to the keystore. However, some keystores do not require the passphrase to read from the keystore. WebLogic Server reads only from the keystore, so whether or not you define this property depends on the requirements of the keystore.
10. Click **Save**.
11. Click **Activate Changes** in the Administration Console's Change Center to make the changes take effect.
12. Select **Configuration**, then **SSL**.
13. Click **Lock & Edit**.
14. In the **Private Key Alias** field, enter the alias you used for the host name the Managed Server listens on.

    In the **Private Key Passphrase** and the **Confirm Private Key Passphrase** fields, enter the password for the keystore that you created in Section 13.3.2, "Creating an Identity Keystore Using the utils.ImportPrivateKey Utility."
15. Click **Save**.
16. Click **Activate Changes** in the Administration Console's Change Center to make the changes take effect.
17. Restart the server for which the changes have been applied.

13.3.6 Changing the Host Name Verification Setting for the Managed Servers

Once the steps above have been performed, you should set host name verification for the affected Managed Servers to `Bea Host Name Verifier`. To do this, perform the following steps:

1. Log in to the WebLogic Server Administration Console.
2. Expand the **Environment** node in the **Domain Structure** tree on the left.

3. Click **Servers**.
4. On the Summary of Servers page, select the Managed Server in the **Names** column of the table.
5. On the settings page for the server, open the **SSL** tab.
6. Expand the **Advanced** section of the page.
7. Click **Lock & Edit**.
8. Set host name verification to `Bea Host Name Verifier`.
9. Click **Save**.
10. Restart the server for which the changes have been applied.

13.4 Starting Node Manager

Run the following commands to start Node Manager on *HOST*. If you have not configured and started Node Manager for the first time yet, run the `setNMProps.sh` script as specified in Section 8.4.2, "Starting Node Manager on SOAHOST1," to enable the use of the start script for your Managed Servers.

```
cd WL_HOME/server/bin

export JAVA_OPTIONS=-DCustomIdentityAlias=appIdentityX

./startNodeManager.sh
```

Notes:

- Make sure to specify the custom identity alias specifically assigned to each host, so `appIdentity1` for ...HOST1 and `appIdentity2` for ...HOST2.
- It is assumed that any other pertaining Java options are passed to Node Manager in each node. For example, in SOAHOST1, which hosts the Administration Server, also include `-DDomainRegistrationEnabled=` along with `-DCustomIdentityAlias=` in the JAVA_OPTIONS parameter.
- Verify that Node Manager is using the appropriate stores and alias from the Node Manager output. Node Manager should prompt out the following:

  ```
  CustomIdentityKeyStoreFileName=ORACLE_BASE/admin/domain_
  name/aserver/domain_name/certs/appIdentityKeyStore.jks
  CustomIdentityAlias=appIdentityX
  ```

  Host name verification works if you apply a test configuration change to the servers, and it succeeds without Node Manager reporting any SSL errors.

14

Configuring Server Migration for an Enterprise Deployment

This chapter describes the procedures for configuring server migration for an enterprise deployment.

This chapter contains the following sections:

14.1 Overview of Server Migration for an Enterprise Deployment

You can configure server migration for Oracle WebLogic Server Managed Servers. With server migration configured, should failure occur, each Managed Server can restart on a different host machine. The Managed Servers listen on specific floating IPs that are failed over by WebLogic Server.

Note: Refer to each component's chapter for details on whether it uses or requires server migration or not.

The procedures described in this chapter must be performed for various components of the enterprise deployment topology outlined in Section 2.1.1, "Reference Topology Documented in the Guide." Variables are used in this chapter to distinguish between component-specific items:

- *WLS_SERVER1* and *WLS_SERVER2* refer to the WebLogic Server Managed Servers for the enterprise deployment component.
- *HOST1* and *HOST2* refer to the host machines for the enterprise deployment component.
- *CLUSTER* refers to the cluster associated with the enterprise deployment component.

The values to be used for these variables are provided in the component-specific chapters in this EDG.

In this enterprise topology, you must configure server migration for the *WLS_SERVER1* and *WLS_SERVER2* Managed Servers. The *WLS_SERVER1* Managed Server is configured to restart on *HOST2* should a failure occur. The *WLS_SERVER2* Managed Server is configured to restart on *HOST1* should a failure occur. For this configuration, the *WLS_SERVER1* and *WLS_SERVER2* servers listen on specific floating IP addresses that are failed over by WebLogic Server migration.

Table 14–1 describes the steps for configuring server migration for the WebLogic Server Managed Servers.

Table 14–1 Steps for Configuring Server Migration

Step	Description	More Information
Set up a user, tablespace, and migration table	Create a `leasing` tablespace, user, and table for server migration.	Section 14.2, "Setting Up a User and Tablespace for the Server Migration leasing Table"
Create GridLink data sources for the `leasing` table	Create a data source for each of the Oracle RAC database instances and the global `leasing` GridLink data source in the Administration Console.	Section 14.3, "Creating a GridLink Data Source for leasing Using the Administration Console"
Specify Node Manager properties values for migration	Edit the property values in the nodemanager.properties file for each host, and verify the values.	Section 14.4, "Editing Node Manager's Properties File"
Set the environment and specify superuser privileges for the oracle user	Add files to the `PATH` environment variable, and grant the `sudo` privilege for the `wlsifconfig.sh` script.	Section 14.5, "Setting Environment and Superuser Privileges for the wlsifconfig.sh Script"
Configure cluster migration	Assign available nodes as migration targets, and specify candidate machines for each server.	Section 14.6, "Configuring Server Migration Targets"
Test server migration	Verify server migration between hosts from Node Manager or the Administration Console.	Section 14.7, "Testing the Server Migration"

14.2 Setting Up a User and Tablespace for the Server Migration leasing Table

Set up a user and tablespace for the server migration `leasing` table using the `create tablespace leasing` command.

> **Note:** If other servers in the same domain have already been configured for server migration, you can use the same tablespace and data sources. In that case, the data sources and GridLink data source for the database table `leasing` do not need to be re-created, but they will have to be retargeted to the cluster being configured with server migration.

To set up a user and tablespace for the server migration leasing table:

1. Create a tablespace called `leasing`. For example, log in to SQL*Plus as the `sysdba` user and run the following command:

```
SQL> create tablespace leasing
       logging datafile 'DB_HOME/oradata/orcl/leasing.dbf'
       size 32m autoextend on next 32m maxsize 2048m extent management local;
```

> **Note:** The database file location will vary depending on the type of storage and data file location used for the database.

2. Create a user named `leasing`, and assign to it the `leasing` tablespace:

```
SQL> create user leasing identified by password;

SQL> grant create table to leasing;

SQL> grant create session to leasing;

SQL> alter user leasing default tablespace leasing;

SQL> alter user leasing quota unlimited on leasing;
```

3. Create the `leasing` table using the `leasing.ddl` script:
   a. Copy the `leasing.ddl` file located in either of the following directories to your database node:

```
WL_HOME/server/db/oracle/817/
WL_HOME/server/db/oracle/920/
```

   b. Connect to the database as the `leasing` user.
   c. Run the `leasing.ddl` script in SQL*Plus:

```
SQL> @copy_location/leasing.ddl;
```

14.3 Creating a GridLink Data Source for leasing Using the Administration Console

Create a GridLink data source for the `leasing` table from the Oracle WebLogic Server Administration Console.

To create a GridLink data source:

1. Log in to the WebLogic Server Administration Console.
2. If you have not already done so, in the **Change Center**, click **Lock & Edit** and click **Next**.
3. In the **Domain Structure** tree, expand **Services**, then select **Data Sources**.
4. On the Summary of Data Sources page, click **New**, select **GridLink Data Source**, and do the following:
   - Enter a logical name for the data source in the **Name** field. For example, `leasing`.
   - Enter a name for **JNDI**. For example, `jdbc/leasing`.
   - For the Database Driver, select **Oracle's Driver (Thin) for GridLink Connections; Versions**: **11 and later**.
   - Click **Next**.
5. In the Transaction Options page, clear **Supports Global Transactions**, and click **Next**.
6. In the GridLink Data Source Connection Properties Options screen, select **Enter individual listener information**, and click **Next**.
7. Enter the following connection properties:
   - **Service Name**: Enter the service name of the database with lowercase characters. For a GridLink data source, you must enter the Oracle RAC service name. For example:

     ```
     wccedg.mycompany.com
     ```

   - **Host Name and Port**: Enter the SCAN address and port for the RAC database being used. You can identify this address by querying the appropriate parameter in the database using the TCP Protocol:

     ```
     SQL>show parameter remote_listener;

     NAME                 TYPE        VALUE

     ------------------------------------------------

     remote_listener     string      db-scan.mycompany.com
     ```

Note: For Oracle Database 11*g* Release 1 (11.1), use the virtual IP and port of each database instance listener; for example:

```
custdbhost1-vip.mycompany.com (port 1521)
```

and

```
custdbhost2-vip.mycompany.com (1521)
```

For Oracle Database 10*g*, use multi data sources to connect to an Oracle RAC database. For information about configuring multi data sources, see Appendix A, "Using Multi Data Sources with Oracle RAC."

- **Port** - The port on which the database server listens for connection requests.
- **Database User Name**: `leasing`
- **Password**: Enter the password for the `leasing` user.
- **Confirm Password**: Enter the password again and click **Next**.

8. On the Test GridLink Database Connection page, review the connection parameters and click **Test All Listeners**. Here is an example of a successful connection notification:

```
Connection test for jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_
LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=db-scan.mycompany.com)
(PORT=1521)))(CONNECT_DATA=(SERVICE_NAME=wccedg.mycompany.com))) succeeded.
```

   Click **Next**.

9. In the ONS Client Configuration page, do the following:
   - Select **FAN Enabled** to subscribe to and process Oracle FAN events.
   - Enter here also the SCAN address for the RAC database and the ONS remote port as reported by the database (example follows) and click **ADD**:

```
[orcl@db-scan1 ~]$ srvctl config nodeapps -s

ONS exists: Local port 6100, remote port 6200, EM port 2016
```

   - Click **Next**.

Note: For Oracle Database 11*g* Release 1 (11.1), use the host name and port of each database's ONS service; for example:

```
custdbhost1.mycompany.com (port 6200)
```

and

```
custdbhost2.mycompany.com (6200)
```

---

10. On the Test ONS Client Configuration page, review the connection parameters, and click **Test All ONS Nodes**.

    Here is an example of a successful connection notification:

    ```
    Connection test for db-scan.mycompany.com:6200 succeeded.
    ```

    Click **Next**.

11. In the Select Targets page, select the targets for which you are doing server migration, **IMG_Cluster** and **SOA_Cluster**, and **All Servers in the cluster**.
12. Click **Finish**.
13. Click **Activate Changes**.

14.4 Editing Node Manager's Properties File

The third step is to edit Node Manager's properties file. This needs to be done for the Node Managers in both nodes where server migration is being configured:

```
Interface=eth0
NetMask=255.255.255.0
UseMACBroadcast=true
```

- **Interface:** This property specifies the interface name for the floating IP (for example, eth0).

  Do not specify the sub-interface, such as `eth0:1` or `eth0:2`. This interface is to be used without `:0` or `:1`. Node Manager's scripts traverse the different `:`*X*-enabled IPs to determine which to add or remove. For example, the valid values in Linux environments are eth0, eth1, eth2, eth3, eth*n*, depending on the number of interfaces configured.

- **NetMask:** This property specifies the net mask for the interface for the floating IP. The net mask should the same as the net mask on the interface; 255.255.255.0 is used as an example in this document.
- **UseMACBroadcast:** This property specifies whether or not to use a node's MAC address when sending ARP packets, that is, whether or not to use the `-b` flag in the `arping` command.

Verify in Node Manager's output (shell where Node Manager is started) that these properties are being used, or problems may arise during migration. You should see something like this in Node Manager's output:

```
...
StateCheckInterval=500
Interface=eth0
NetMask=255.255.255.0
...
```

> **Note:** The following step is not required if the server properties (start properties) have been properly set and Node Manager can start the servers remotely.

- Set the following property in the `nodemanager.properties` file:

  StartScriptEnabled: Set this property to `true`. This is required for Node Manager to start the Managed Servers using start scripts.

Note: When you run Node Manager from a shared storage installation, multiple nodes are started using the same nodemanager.properties file. However, each node may require different NetMask or Interface properties. In this case, specify individual parameters on a per-node basis using environment variables. For example, to use a different interface (eth3) in *HOSTn*, use the Interface environment variable as follows:

```
export JAVA_OPTIONS=-DInterface=eth3
```

Start Node Manager after the variable has been set in the shell.

14.5 Setting Environment and Superuser Privileges for the wlsifconfig.sh Script

The fourth step is to set environment and superuser privileges for the wlsifconfig.sh script (for the `oracle` user):

1. Ensure that your `PATH` environment variable includes the files listed in Table 14–2.

Table 14–2 Files Required for the PATH Environment Variable

File	Located in This Directory
`wlsifconfig.sh`	*ORACLE_BASE*`/admin/`*domain_name*`/mserver/`*domain_name*`/bin/server_migration/`
`wlscontrol.sh`	*WL_HOME*`/common/bin/`
`nodemanager.domains`	*WL_HOME*`/common/nodemanager/`

2. Grant the `sudo` privilege for the `wlsifconfig.sh` script.
   - Configure `sudo` to work without a password prompt.
   - For security reasons, `sudo` should be restricted to the subset of commands required to run the `wlsifconfig.sh` script. For example, perform these steps to set the `environment` and `superuser` privileges for the `wlsifconfig.sh` script:
     a. Grant the `sudo` privilege to the WebLogic Server user (`oracle`) with no password restriction, and grant `execute` privilege on the `/sbin/ifconfig` and `/sbin/arping` binaries.
     b. Make sure the script is executable by the WebLogic Server user (`oracle`). The following is an example of an entry inside `/etc/sudoers` granting the `sudo` execution privilege for `oracle` and also over `ifconfig` and `arping`:

```
oracle ALL=NOPASSWD: /sbin/ifconfig,/sbin/arping
```

Note: Ask the system administrator for the `sudo` and `system` privileges as appropriate for this step.

3. Start Node Manager on *HOST1* and *HOST2* by running the `startNodeManager.sh` script, which is located in the *WL_HOME*`/server/bin/` directory.

14.6 Configuring Server Migration Targets

The fifth step is to configure server migration targets. You first assign all the available nodes for the cluster's members and then specify candidate machines (in order of preference) for each server that is configured with server migration. Follow these steps to configure cluster migration in a migration in a cluster:

1. Log in to the WebLogic Server Administration Console (`http://`*`Host`*`:`*`Admin_Port`*`/console`). Typically, *`Admin_Port`* is `7001` by default.
2. In the **Domain Structure** tree on the left, expand **Environment** and select **Clusters**.
3. On the Summary of Clusters page, click the cluster for which you want to configure migration (***`CLUSTER`***) in the **Name** column of the table.

   Note: For the procedures in this document, configure server migration for the Oracle SOA Suite and Imaging clusters.

4. Open the **Migration** tab.
5. Click **Lock & Edit**.
6. In the **Available** field, select the machine to which to allow migration and click the right arrow. In this case, select ***`HOST1`*** and ***`HOST2`***.
7. Select the data source to be used for automatic migration. In this case, select the `leasing` data source.
8. Click **Save**.
9. Click **Activate Changes**.
10. Click **Lock & Edit**.
11. Set the candidate machines for server migration. You must perform this task for all of the Managed Servers as follows:
    a. In the **Domain Structure** tree on the left of the WebLogic Server Administration Console, expand **Environment** and select **Servers**.
    b. Select the server for which you want to configure migration.

       Note: For the procedures in this document, configure server migration for the Oracle SOA Suite and Imaging servers.

    c. Open the **Migration** tab.
    d. In the **Available** field, located in the Migration Configuration section, select the machines to which to allow migration and click the right arrow. For ***`WLS_SERVER1`***, select ***`HOST2`***. For ***`WLS_SERVER2`***, select ***`HOST1`***.
    e. Select **Automatic Server Migration Enabled**. This enables Node Manager to start a failed server on the target node automatically.
    f. Click **Save**.
    g. Click **Activate Changes**.
    h. Restart the Administration Server, node managers, and the servers for which server migration has been configured.

14.7 Testing the Server Migration

The sixth and final step is to test the server migration. To verify that server migration is working properly:

From *HOST1*:

1. Stop the WLS_SERVER1 Managed Server. To do this, run this command on *HOST1*:

   ```
   kill -9 pid
   ```

   where *pid* specifies the process ID of the Managed Server. You can identify the process ID in the node by running this command:

   ```
   ps -ef | grep WLS_SERVER1
   ```

2. Watch the Node Manager console. You should see a message indicating that *WLS_SERVER1*'s floating IP has been disabled.
3. Wait for Node Manager to try a second restart of *WLS_SERVER1*. It waits for a fence period of 30 seconds before trying this restart.
4. Once Node Manager restarts the server, stop it again. Node Manager should now log a message indicating that the server will not be restarted again locally.

From *HOST2*:

1. Watch the local Node Manager console. After 30 seconds since the last try to restart *WLS_SERVER1* on node 1, Node Manager on node 2 should prompt that the floating IP for *WLS_SERVER1* is being brought up and that the server is being restarted in this node.
2. Access your server's console in the same IP.

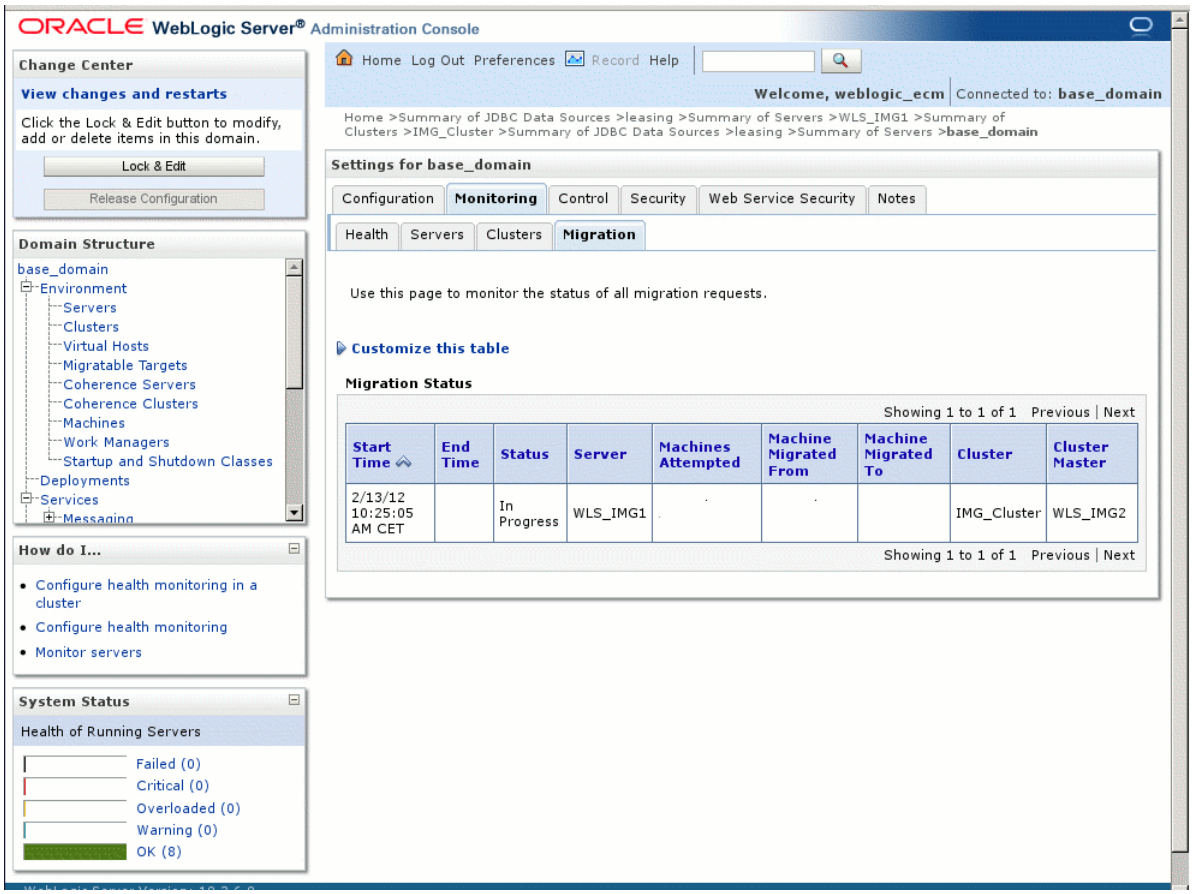
Verification from the Administration Console

Migration can also be verified in the Administration Console:

1. Log in to the Administration Console.
2. Click **Domain** on the left.
3. Click the **Monitoring** tab and then the **Migration** subtab.

   The Migration Status table provides information on the status of the migration (Figure 14–1).

Figure 14–1 Migration Status Screen in the Administration Console



Note: After a server is migrated, to fail it back to its original node or machine, stop the Managed Server from the WebLogic Server Administration Console and then start it again. The appropriate Node Manager will start the Managed Server on the machine to which it was originally assigned.

15

Integrating with Oracle Identity Management

This chapter describes how to integrate Oracle WebCenter Content with Oracle Identity Management. It contains the following sections:

15.1 Overview of Integrating with Oracle Identity Management

You can integrate an Oracle Fusion Middleware enterprise deployment with Oracle Identity Management 10*g* or 11*g*. The following sections describe how to first configure credential and policy stores, reassociate those credential and policy stores, and then integrate with Oracle Identity Management 10*g* or 11*g*.

> **Note:** When integrating with Oracle Identity Management, use the transport mode currently in use by the Oracle Identity Management servers. For example, Open, Simple, or Cert.

Table 15–1 lists the high-level steps for integrating Oracle Identity Management 10*g* with an Oracle WebCenter Content enterprise deployment.

Table 15–1 Steps for Integrating with Oracle Identity Management 10g

Step	Description	More Information
Configure the credential store	Configure the Oracle Internet Directory LDAP as a credential store for the Oracle WebCenter Content enterprise deployment topology.	Section 15.2, "Credential and Policy Store Configuration"
Configure the policy store	Configure the Oracle Internet Directory LDAP as the policy store for the Oracle WebCenter Content enterprise deployment topology.	Section 15.2.3, "Policy Store Configuration"
Run the OAM Configuration Tool	The OAM Configuration Tool (oamcfg) starts a series of scripts and sets up the required policies.	Section 15.3.3.2, "Running the OAM Configuration Tool"
Install and configure WebGate	Install WebGate on each of the WEBHOST*n* machines to secure the Web tier.	Section 15.3.4, "Installing and Configuring WebGate"
Configure IP validation for the Webgate	Configure the IP validation for the Webgate using Access System Console.	Section 15.3.5, "Configuring IP Validation for the Enterprise Deployment Webgate"
Set up WebLogic Server authenticators	Set up the WebLogic Server authenticators by backing up the configuration files, setting up the Oracle Access Manager ID Asserter, and setting the order of providers.	Section 15.3.6, "Setting up the Oracle Access Manager Identity Asserter"

Table 15–2 lists the high-level steps for integrating Oracle Identity Management 11*g* with an Oracle WebCenter Content enterprise deployment.

Table 15–2 Steps for Integrating with Oracle Identity Management 11g

Step	Description	More Information
Configure the credential store	Configure the Oracle Internet Directory LDAP as a credential store for the Oracle WebCenter Content Enterprise Deployment topology.	Section 15.2, "Credential and Policy Store Configuration"
Configure the policy store	Configure the Oracle Internet Directory LDAP as the policy store for the Oracle WebCenter Content Enterprise Deployment topology.	Section 15.2.3, "Policy Store Configuration"
Install WebGate	Install WebGate on each of the WEBHOST machines where an HTTP Server has already been installed.	Section 15.3.4, "Installing and Configuring WebGate"
Register the WebGate agent	Register the Webgate agent using the RREG tool.	Section 15.4.4, "Registering the WebGate Agent"
Set up WebLogic Server authenticators	Set up the WebLogic Server authenticators by backing up the configuration files, setting up the Oracle Access Manager ID Asserter, and setting the order of providers.	Section 15.3.6, "Setting up the Oracle Access Manager Identity Asserter"

15.2 Credential and Policy Store Configuration

The following topics describe credential and policy store configuration in detail:

- Section 15.2.1, "Overview of Credential and Policy Store Configuration"
- Section 15.2.2, "Credential Store Configuration"
- Section 15.2.3, "Policy Store Configuration"
- Section 15.2.4, "Reassociation of Credentials and Policies"

15.2.1 Overview of Credential and Policy Store Configuration

Oracle Fusion Middleware allows using different types of credential and policy stores in an Oracle WebLogic Server domain. Domains can use stores based on Oracle Database, 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 enterprise deployment topology for Oracle WebCenter Content uses different domain homes for the Administration Server and the Managed Servers, which means that Oracle requires the use of an LDAP store as policy and credential store for integrity and consistency. By default, 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 and 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. For more information, see Section 15.2.4, "Reassociation of Credentials and Policies."

15.2.2 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. This section provides steps to configure Oracle Internet Directory LDAP as a credential store for the Oracle WebCenter Content enterprise deployment topology. For more details on credential store configuration, see "Configuring the Credential Store" in the *Oracle Fusion Middleware Security Guide*.

The following sections describe credential store configuration:

- Section 15.2.2.1, "Creating the LDAP Authenticator"
- Section 15.2.2.2, "Setting the Order of Providers"
- Section 15.2.2.3, "Moving the WebLogic Server Administrator to LDAP"
- Section 15.2.2.4, "Reassociating the Domain Credential Store"

15.2.2.1 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:

```
ORACLE_BASE/admin/domain_name/aserver/domain_name/servers/AdminServer/security/
boot.properties
```

Follow these steps to set the proper authenticator:

1. Log in to the Oracle WebLogic Server Administration 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 **Lock & Edit**.
7. Click the **New** button to add a new provider.
8. Enter a name for the provider such as **OIDAuthenticator** or **OVDAuthenticator** depending on whether Oracle Internet Directory or Oracle Virtual Directory will be used.
9. Select the **OracleInternetDirectoryAuthenticator** or **OracleVirtualDirectoryAuthenticator** type from the list of authenticators depending on whether Oracle Internet Directory or Oracle Virtual Directory will be used.
10. Click **OK**.
11. In the Providers screen, click the newly created Authenticator.
12. 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**.
13. Click **Save** to save this setting.
14. Open the **Provider Specific** tab to enter the details for the LDAP server.
15. 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.

16. Click **Activate Changes** to propagate the changes.

15.2.2.2 Setting the Order of Providers

Reorder the OID/OVD Authenticator and Default Authenticator and ensure that the control flag for each authenticator is set in the following order:

- OID LDAP Authenticator: **SUFFICIENT**
- Default Authenticator: **SUFFICIENT**

To set the order of providers:

1. Log in to the WebLogic Server Administration Console, if not already logged in.
2. Click **Lock & Edit.**
3. Navigate to **SecurityRealms**, then the default realm name, and then **Providers**.
4. Click **Reorder** and ensure that the control flags for the OID/OVD Authenticator, and Default Authenticator are set as follows:
   - OID LDAP Authenticator (or OVD LDAP Authenticator): **SUFFICIENT**
   - Default Authenticator: **SUFFICIENT**
5. Click **OK**.
6. Click **Activate Changes** to propagate the changes.
7. Restart the Administration Server and all Managed Servers.

15.2.2.3 Moving the WebLogic Server Administrator to LDAP

This section provides details for provisioning a new administrator user and group for managing the Oracle WebCenter Content WebLogic Server domain in the enterprise deployment topology. This section describes the following tasks:

- Section 15.2.2.3.1, "Provisioning Admin Users and Groups in an LDAP Directory"
- Section 15.2.2.3.2, "Assigning the Admin Role to the Admin Group"
- Section 15.2.2.3.3, "Updating the boot.properties File and Restarting the System"

15.2.2.3.1 Provisioning Admin Users and Groups in an LDAP Directory As mentioned in the introduction to this section, users and groups from multiple WebLogic Server domains may be provisioned in a central LDAP user store. In such a case, there is a possibility that one WebLogic Server administration 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 administration user and group for the Oracle WebCenter Content WebLogic Server domain will be provisioned with the following DNs:

- Admin User DN:

  ```
  cn=weblogic_ecm,cn=Users,dc=us,dc=mycompany,dc=com
  ```

- Admin Group DN:

  ```
  cn=ECM 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 following contents, depending on the Oracle Access Manager version used, and then save the file:

   - **Oracle Access Manager 10*g*:**

     ```
     dn: cn=weblogic_ecm, cn=Users, dc=us, dc=mycompany, dc=com
     orclsamaccountname: weblogic_ecm
     givenname: weblogic_ecm
     sn: weblogic_ecm
     userpassword: password
     mail: weblogic_ecm
     objectclass: top
     objectclass: person
     objectclass: organizationalPerson
     objectclass: inetorgperson
     objectclass: orcluser
     objectclass: orcluserV2
     uid: weblogic_ecm
     cn: weblogic_ecm
     description: Admin User for the Oracle WebCenter Content Domain
     ```

   - **Oracle Access Manager 11*g*:**

     ```
     dn: cn=weblogic_ecm, cn=Users, dc=us, dc=mycompany, dc=com
     orclsamaccountname: weblogic_ecm
     givenname: weblogic_ecm
     sn: weblogic_ecm
     userpassword: password
     mail: weblogic_ecm
     objectclass: top
     objectclass: person
     ```

```
objectclass: organizationalPerson
objectclass: inetorgperson
objectclass: orcluser
objectclass: orcluserV2
uid: weblogic_ecm
cn: weblogic_ecm
description: Admin User for the Oracle WebCenter Content Domain
```

2. Run the `ldapadd` command on the Oracle Internet Directory host 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 Oracle Identity Management installation where Oracle Internet Directory resides.

   For example (all on a single line):

   ```
   ORACLE_HOME/bin/ldapadd -h oid.mycompany.com -p 389 -D cn="orcladmin"
   -w password -c -v -f admin_user.ldif
   ```

3. Create an `LDIF` file named `admin_group.ldif` with the following contents, and then save the file:

   ```
   dn: cn=ECM Administrators, cn=Groups, dc=us, dc=mycompany, dc=com
   displayname: ECM Administrators
   objectclass: top
   objectclass: groupOfUniqueNames
   objectclass: orclGroup
   uniquemember: cn=weblogic_ecm, cn=users, dc=us, dc=mycompany, dc=com
   cn: ECM Administrators
   description: Administrators Group for the Oracle WebCenter Content Domain
   ```

4. Run the `ldapadd` command on the Oracle Internet Directory host located under the *ORACLE_HOME*`/bin/` directory to provision the group in Oracle Internet Directory (all on a single line):

   ```
   ORACLE_HOME/bin/ldapadd -h oid.mycompany.com -p 389 -D cn="orcladmin"
   -w password -c -v -f admin_group.ldif
   ```

15.2.2.3.2 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 Server 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 in to 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, open 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 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 **ECM 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 `ECM Administrators Group` as an entry.
9. Click **Save** to finish adding the `Admin Role` to the `ECM Administrators Group`.
10. Validate that the changes were successful by bringing up the WebLogic Server Administration Console using a web browser. Log in using the credentials for the `weblogic_ecm` user.

> **Note:** Each Oracle application in the Oracle WebCenter Content enterprise deployment topology may have its own predefined roles and groups defined for administration and monitoring purposes. By default, the `Administrators` group will allow these operations. However, this group may be too broad. For example, it may be undesirable that Oracle SOA Suite administrators are also administrators for the WebLogic Server domain where Oracle SOA Suite, Oracle WebCenter Content, and Oracle WebCenter Content: Imaging are running. This is why it may be desirable, as suggested in this section, to create a more specific group such as `ECM Administrators`. For the various applications to allow the `ECM Administrators` group to administer the different systems, you need to add the required roles to that group. For example, for SOA Worklistapp's administration, add the SOAAdmin role. Refer to each component's specific roles for the required roles in each case.

15.2.2.3.3 Updating the boot.properties File and Restarting the System

The `boot.properties` file for the Administration Server should be updated with the WebLogic Server administration user created in Oracle Internet Directory. Follow these steps to update the `boot.properties` file:

1. On SOAHOST1, go the following directory:

   ```
   cd ORACLE_BASE/admin/domain_name/aserver/domain_name/servers/AdminServer/
   security
   ```

2. Rename the existing boot.properties file:

   ```
   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=weblogic_ecm
   password=password
   ```

4. Save the file.
5. Stop the Administration Server using the following command:

```
wls:/nm/domain_name>nmKill("AdminServer")
```

6. Start the Administrator Server using the procedure in Section 8.4.3, "Starting the Administration Server on SOAHOST1."

15.2.2.4 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 15.2.4, "Reassociation of Credentials and Policies" for detailed steps.

15.2.3 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 WebCenter Content enterprise deployment topology. For more details on policy store configuration, refer to "OPSS Authorization and the Policy Store" in the *Oracle Fusion Middleware Security Guide*.

15.2.3.1 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_ecm
cn: jpsroot_ecm
objectclass: top
objectclass: OrclContainer
```

   The distinguished name of the root node (illustrated by the string `jpsroot_ecm` above) must be distinct from any other distinguished name. One root node can be shared by multiple WebLogic Server 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 the Oracle Internet Directory server using the `ldapadd` command on the Oracle Internet Directory server, as illustrated in the following example (all on a single line):

```
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 `ldapsearch` command on the Oracle Internet Directory server, as illustrated in the following example (all on a single line):

```
ORACLE_HOME/bin/ldapsearch -h ldap_host -p ldap_port -D cn=orcladmin
-w password -b "cn=jpsroot_ecm" objectclass="orclContainer"
```

4. When using Oracle Internet Directory as the LDAP-based policy store, run the `oidstats.sql` utility in the INFRADBHOST servers to generate database statistics for optimal database performance:

   ```
   ORACLE_HOME/bin/sqlplus
   ```

   Enter ODS as a user name. You will be prompted for credentials for the ODS user. Inside SQL*Plus, enter the command to gather the statistics info:

   ```
   @ORACLE_HOME/ldap/admin/oidstats.sql
   ```

   The `oidstats.sql` utility must be run just once after the initial provisioning. For details about this utility, see "Oracle Internet Directory Administration Tools" in the *Oracle Fusion Middleware Reference for Oracle Identity Management*.

15.2.3.2 Reassociating the Domain Policy Store

Reassociating the policy store consists in migrating policy data from a file-based 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-based 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 Oracle Enterprise Manager Fusion Middleware Control or the WLST `reassociateSecurityStore` command. See Section 15.2.4, "Reassociation of Credentials and Policies" for detailed steps.

15.2.4 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:

   ```
   cd ORACLE_COMMON_HOME/common/bin

   ./wlst.sh
   ```

2. Connect to the WebLogic Server Administration Server using the following `wlst connect` command.

   Syntax:

   ```
   connect("Admin_User","Admin_User_Password","t3://hostname:port")
   ```

   For example:

   ```
   connect("weblogic","password","t3://ADMINVHN:7001")
   ```

3. Run the `reassociateSecurityStore` command, as follows:

   Syntax:

   ```
   reassociateSecurityStore(domain="domain_name",admin="cn=orcladmin",
   password="orclPassword",ldapurl="ldap://LDAP_HOST:LDAP_PORT",servertype="OID",
   jpsroot="cn=jpsroot_ecm")
   ```

   For example:

   ```
   wls:/domain_name/serverConfig>reassociateSecurityStore(domain="domain_name",
   admin="cn=orcladmin",password="password",ldapurl="ldap://oid.mycompany.com:389"
   ,servertype="OID",jpsroot="cn=jpsroot_ecm")
   ```

   The output for the command is shown below:

   ```
   {servertype=OID,jpsroot_ecm=cn=jpsroot_ecm_idm_idmhost1,admin=cn=orcladmin,
   domain=IDMDomain,ldapurl=ldap://oid.mycompany.com:389,password=password}
   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 the 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
   Starting keystore reassociation
   The server and ServiceConfigurator setup done.
   Schema is seeded into the server
   Data is migrated to the server
   Service in the server after migration has been tested to be available
   Update of jps configuration is done
   keystore reassociation done
   Jps Configuration has been changed. Please restart the server.
   ```

4. Restart the Administration Server after the command completes successfully.

   To restart the Administration Server, use the procedure in Section 8.4.3, "Starting the Administration Server on SOAHOST1."

> **Note:** For credential and policy changes to take effect, the servers in the domain must be restarted.

15.2.4.1 Cataloging Oracle Internet Directory Attributes

An Oracle Internet Directory attribute used in a search filter must be indexed. The indexing is an optional procedure used to enhance performance. If not done yet in this OID, use the `catalog` tool to index attributes:

```
catalog connect="orcl" add=true attribute="orclrolescope" verbose="true"
```

Optionally, the attribute names can be placed in a file and processed in a batch as follows:

```
orclrolescope
orclassignedroles
orclApplicationCommonName
orclAppFullName
orclCSFAlias
orclCSFKey
orclCSFName
orclCSFDBUrl
orclCSFDBPort
orclCSFCredentialType
orclCSFExpiryTime
modifytimestamp
createtimestamp
orcljpsassignee
```

For more information on indexing OID attributes, see "Tasks and Examples for catalog" in the *Oracle Fusion Middleware Reference for Oracle Identity Management*.

15.3 Oracle Access Manager 10g Integration

This section describes how to set up Oracle Access Manager 10*g* as the single sign-on solution for the Oracle WebCenter Content enterprise deployment topology. It contains the following sections:

- Section 15.3.1, "Overview of Oracle Access Manager Integration"
- Section 15.3.2, "Prerequisites for Oracle Access Manager"
- Section 15.3.3, "Configuring Oracle Access Manager"
- Section 15.3.4, "Installing and Configuring WebGate"
- Section 15.3.5, "Configuring IP Validation for the Enterprise Deployment Webgate"
- Section 15.3.6, "Setting up the Oracle Access Manager Identity Asserter"

15.3.1 Overview of Oracle Access Manager Integration

Oracle Access Manager is the recommended single sign-on (SSO) solution for Oracle Fusion Middleware 11*g* Release 1. For more information on installing and configuring an Oracle Access Manager installation, see *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*. This section explains the procedure for configuring the Oracle WebCenter Content installation with an existing Oracle Access Manager 10*g* installation and the underlying directory service. Oracle recommends using either Oracle Internet Directory or Oracle Virtual Directory or both of these directory services.

Note: The Oracle WebCenter Content enterprise deployment topology described in this book uses a single sign-on configuration where both the Oracle WebCenter Content system and the single sign-on system are in the same network domain (`mycompany.com`). For a multidomain configuration, see the required configuration steps in "Introduction to the OAM Policy Model, Single Sign-On" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager with Oracle Security Token Service*.

15.3.2 Prerequisites for Oracle Access Manager

The setup for Oracle Access Manager assumes an existing Oracle Access Manager 10*g* installation complete with Access Managers and a policy protecting the Policy Manager. For more information on installing and configuring an Oracle Access Manager installation, see the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*. This setup includes a directory service such as Oracle Internet Directory, either standalone or as part of an Oracle Virtual Directory configuration.

In addition, the Oracle Access Manager installation should have its own Web server configured with WebGate. This section also provides the steps for using the Oracle Access Manager Web server as a delegated authentication server.

15.3.3 Configuring Oracle Access Manager

This section covers the following topics:

15.3.3.1 Collecting the Information for the OAM Configuration Tool

The OAM Configuration Tool (oamcfg) starts a series of scripts and sets up the required policies for Oracle Access Manager. It requires various parameters as inputs. Specifically, it creates the following:

1. A form authentication scheme in Oracle Access Manager
2. Policies to enable authentication in WebLogic Server
3. A WebGate entry in Oracle Access Manager 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. A host identifier, depending on the scenario chosen (a default host identifier would be used, if not provided)
6. Policies to protect and unprotect application specific URLs.

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 a high-availability or enterprise deployment 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_aaa_host:** host name of an Oracle Access Manager.
7. **oam_aaa_port:** port of the Oracle Access Manager.

15.3.3.2 Running the OAM Configuration Tool

Before running the OAM Configuration Tool, you must first add the required resources to Oracle Access Manager 10*g* for the Oracle WebCenter Content components. Create a file containing the list of URIs you want to protect with the following content:

```
########################################
#Product Name: Oracle WebCenter Content
########################################
########################################
protected_uris
########################################
/adfAuthentication
/imaging/faces
/em
/console
/DefaultToDoTaskFlow
/sdpmessaging/userprefs-ui
/integration/worklistapp
/workflow/sdpmessagingsca-ui-worklist
/soa/composer
/soa-infra/deployer
/soa-infra/events/edn-db-log
/soa-infra/cluster/info

#"Policy using Basic Authn Scheme" is the name of the policy
#"Basic Over LDAP" is the authentication scheme configured for this #policy
# Note that the name of the policy and the scheme name in the URIs file
# is tab-separated. In other words, there must be a tab between
# "Basic Authn Scheme" and "OraDefaultBasicAuthNScheme" below.
Policy using Basic Authn Scheme OraDefaultBasicAuthNScheme
/inspection.wsil
```

```
########################################
public_uris
########################################

/soa-infra/services
/soa-infra/directWSDL
```

> **Note:** In Oracle Access Manager 10*g*, all resources under a URL prefix are protected by the default rules of a policy domain unless more specific rules are applied to them through policies. For details on the different patterns you can use if you need more specialized protection patterns, see the *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager with Oracle Security Token Service*.

The OAM Configuration Tool resides in the *ORACLE_COMMON_HOME*/modules/oracle.oamprovider_11.1.1/ directory (*ORACLE_COMMON_HOME* depends on the machine where you are running the configuration tool). The tool can be run from any machine with the required installation files. In this case, we run it from WCCHOST1. Run the OAM Configuration Tool for Oracle Access Manager 10*g* registration as follows (all on a single command line):

```
MW_HOME/jrockit_160_version/bin/java -jar oamcfgtool.jar mode=CREATE
app_domain="ECM_EDG"
uris_file="full_path_to_file_containing_uri_definitions"
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=OAMPORT1
```

If your command ran successfully, you should see the following output:

```
Date,Time oracle.security.oam.oamcfg.OAMCfgGlobalConfigHandler
constructGlobalConfig
INFO: Processed input parameters
May 9, 2011 5:09:40 AM oracle.security.oam.oamcfg.OAMCfgGlobalConfigHandler
constructGlobalConfig
INFO: Initialized Global Configuration
Date,Time oracle.security.oam.oamcfg.create.impl.OAMCfgConfigCreator doCreate
INFO: Successfully completed the Create operation.
Date,Time oracle.security.oam.oamcfg.create.impl.OAMCfgConfigCreator doCreate
INFO:  Operation Summary:
Date,Time oracle.security.oam.oamcfg.create.impl.OAMCfgConfigCreator doCreate
INFO:      Policy Domain  : ECM_EDG
Date,Time oracle.security.oam.oamcfg.create.impl.OAMCfgConfigCreator doCreate
INFO:      Host Identifier: ECM_EDG
Date,Time oracle.security.oam.oamcfg.create.impl.OAMCfgConfigCreator doCreate
INFO:      Access Gate ID : ECM_EDG_AG
```

15.3.3.3 Oracle Access Manager Logout Guidelines

For applications invoked by Oracle WebCenter Content and Oracle WebCenter Content: Imaging to comply with Oracle Access Manager logout guidelines (in particular, applications that invoke a logout through `/adfAuthentication?logout=true&end_url=some_URI`), integration with an Oracle Access Manager 10*g* environment requires additional configuration on the WebGate to handle the `end_url`. Without this additional configuration, you are logged out, but not redirected to the end URL because Oracle Access Manager 10*g* WebGate does not process `end_url`. For information about configuration procedures, see the *Oracle Fusion Middleware Security Guide*.

When integrating Oracle WebCenter Content with Oracle Access Manager 10*g*, you must add the URL `/oamsso/logout.html` to the logout URL setting for the Access Gate for the single sign-on logout to work properly. For more information, see "Configuring a Single Sign-On Logout URL" and "AccessGate Configuration Parameters" in the *Oracle Access Manager Access Administration Guide*.

15.3.3.4 Verifying Successful Creation of the Policy Domain and AccessGate

Verifying the Policy Domain

To verify the policy domain:

1. Log in to 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, which includes the domain you just created. It will have the suffix _PD (for example, ECM_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 to go to the General area of this domain.
5. Open the **Resources** tab and 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:

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. Open the **Access System Configuration** tab.
3. Click the **AccessGate Configuration** link on the left panel.
4. Enter `ECM_EDG` as the search criterion (or any other substring you may have used as the `app_domain` name in Section 15.3.3.2, "Running the OAM Configuration Tool"), and click **Go**.
5. Once the access gate for the domain you just created shows up (this will have the suffix _AG (for example, `ECM_EDG_AG`), click it, and you will see the details of the access gate you just created.

15.3.3.5 Verifying That the Cookieless Basic Authorization Scheme Has Been Properly Assigned

To verify that the cookieless basic authorization scheme has been properly assigned:

1. Log in to 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 that have been created.
4. Click **ECM_EDG**.
5. Open the **Policies** tab and then click **Policy using Basic Authn Scheme**.
6. Open the **General** section.

   The inspection.wsil resource should be listed.

7. Open the **Authentication Rule** section.

   The OraDefaultBasicAuthNScheme authentication scheme should be listed.

15.3.3.6 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.

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://host_name:port/access/oblix`

   where *`host_name`* refers to the host where the 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 user name 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, open 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 the host identifier created by the OAM Configuration Tool (for example, **ECM_EDG**).
7. On the Host Identifier Details page, click **Modify**.
8. Add the **Preferred HTTP Host** value used in the Access System Configuration. The following is a list of all the possible host name variations using SSO/ WebGate:

   ```
   webhost1.mydomain.com:7777
   webhost2.mydomain.com:7777
   soahost1vhn1.mycompany.com:8001
   soahost2vhn1.mycompany.com:8001
   soainternal.mycompany.com:80
   ```

```
wcchost1vhn1.mycompany.com:16000
wcchost2vhn1.mycompany.com:16000
wcchost1.mycompany.com:16200
wcchost2.mycompany.com:16200
wcc.mycompany.com:443
admin.mycompany.com:80
adminvhn.mycompany.com:7001
sso.mycompany.com:7779 [WebGate access with Oracle Oracle Identity Management
port]
```

9. Select the **Update Cache** checkbox, 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.

15.3.3.7 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.

To update the WebGate profile created by the OAM Configuration Tool:

1. Navigate to the Access System Console by specifying the following URL in your web browser:

   `http://`*`host_name`*`:`*`port`*`/access/oblix`

   where *`host_name`* refers to the host where the 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 **Access System Configuration**, and then click the Access Gate Configuration Link in the left pane to display the AccessGates Search page.

4. Enter the proper search criteria and click **Go** to display a list of access gates.

5. Select the access gate created by the OAM Configuration Tool (for example, **ECM_EDG_AG**).

6. On the AccessGate Details page, select **Modify** to display the Modify AccessGate page.

7. On the Modify AccessGate page, update the following:

   - **Hostname**: Update the host name 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 host name variations specified in the previous section, for example: `admin.mycompany.com:80`.
   - **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 following message is displayed: `Are you sure you want to commit these changes?`

   Click **OK** to finish updating the configuration.

9. Verify the values displayed on the Details for AccessGate page to confirm that the updates were successful.

15.3.3.8 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 (**ECM_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.

15.3.3.9 Configuring Delegated Form Authentication

To configure the form authentication to redirect to the WebGate that was installed with the Oracle Access Manager installation:

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 Oracle Identity Management installation; for example: `http://sso.mycompany.com`. Click **Save** when you are done.

A WebGate should already be installed in the Oracle Identity Management installation. For details, see the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*.

15.3.4 Installing and Configuring WebGate

WebGate needs to be installed on both WEBHOST1 and WEBHOST2 to secure the web tier:

Note: There is a known issue with the Oracle Access Manager installer that sometimes manifests as a hang at install time on Linux. This is a third-party issue caused by InstallShield. To work around this issue, follow these steps:

1. Copy and paste the following in the shell where you start the installer:

```
cd /tmp
mkdir bin.$$
cd bin.$$
cat > mount <<EOF
#! /bin/sh
exec /bin/true
EOF
chmod 755 mount
export PATH=`pwd`:$PATH
```

2. Run the installation.
3. When the installer is finished running, clean the temporary directory using this command:

```
rm -r /tmp/bin.$$
```

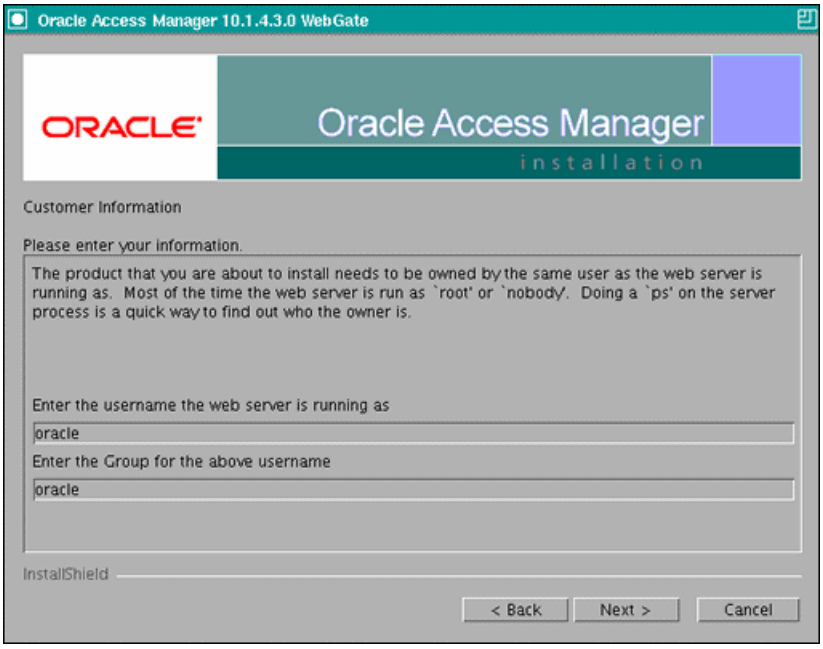
---

1. Launch the WebGate installer (see Section 2.4, "Software Components to Install," for information on where to obtain it) using the following command on WEBHOST*n*:

```
./Oracle_Access_Manager10_1_4_3_0_linux_OHS11g_WebGate -gui
```

2. In the Welcome screen, click **Next**.
3. In the Customer Information screen (Figure 15–1), enter the user name and user group that the web server is running as. Click **Next** to continue.

Figure 15–1 Customer Information Screen

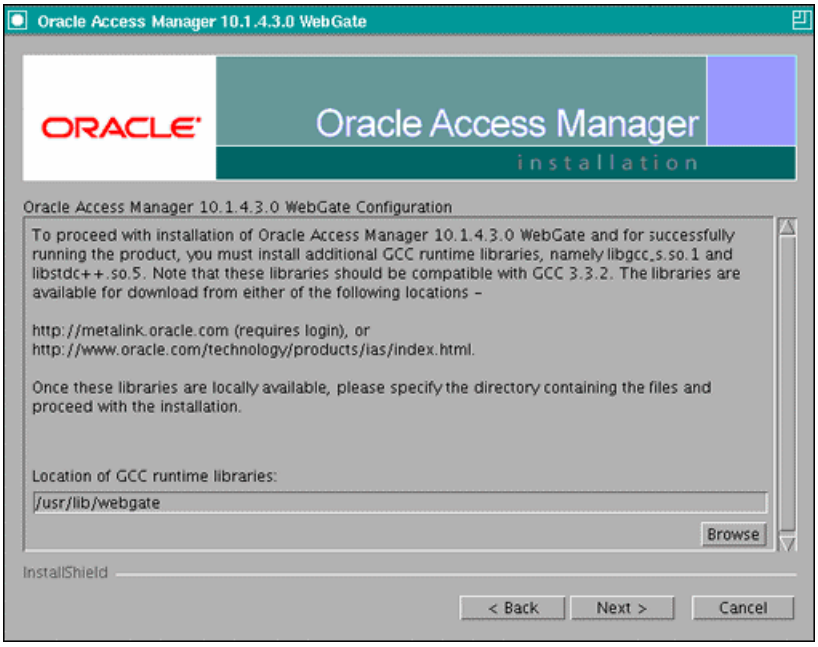


4. In the installation target screen (Figure 15–2), specify the directory where WebGate should be installed. Click **Next** to continue.

Figure 15–2 Installation Target Screen

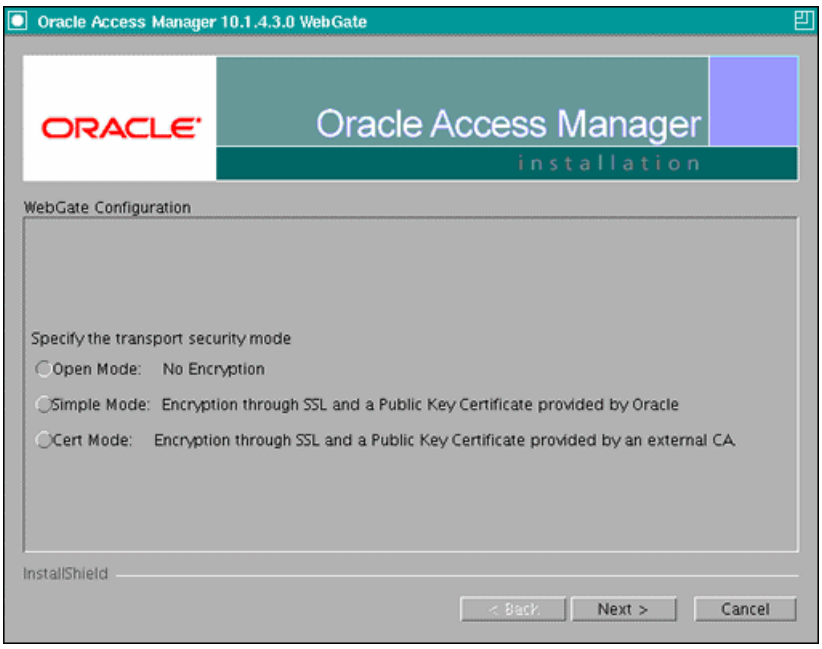
5. In the installation summary screen, click **Next**.
6. Download the required GCC runtime libraries for WebGate as instructed in the WebGate configuration screen (Figure 15–3), and use **Browse** to point to their location on the local computer. Click **Next** to continue.

Figure 15–3 Runtime Libraries Screen



7. The installer now creates the required artifacts. After that is completed, click **Next** to continue.
8. In the transport security mode screen (Figure 15–4), select **Open Mode: No Encryption** and click **Next** to continue.

Figure 15–4 Transport Security Mode Screen



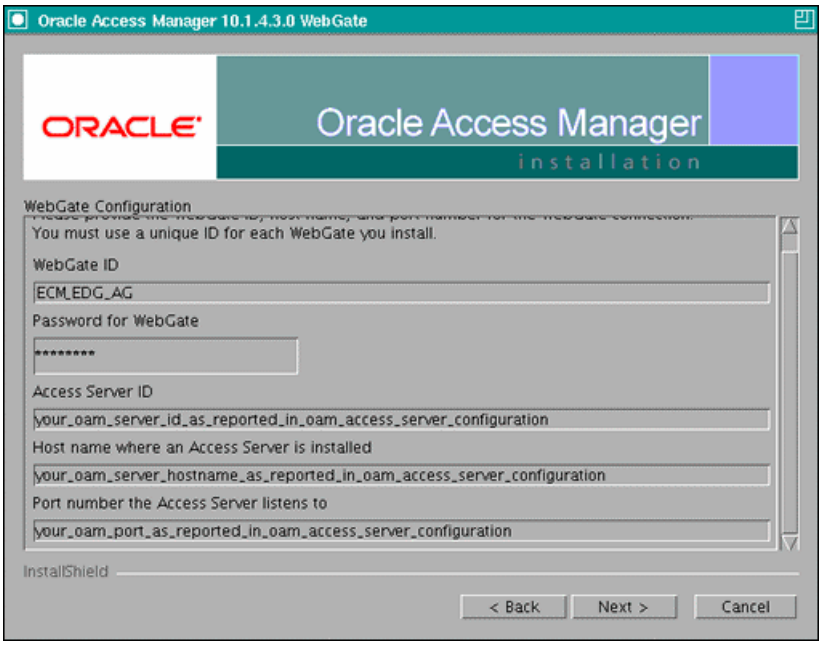
9. In the WebGate configuration screen, provide the details of the access server that will be used. You must provide the following information:

   - **WebGate ID**, as provided when the OAM Configuration Tool was executed
   - **Password for WebGate**
   - **Access Server ID**, as reported by the Oracle Access Manager Access Server configuration
   - **Access Server host name**, as reported by the Oracle Access Manager Access Server configuration
   - **Access Server port number**, as reported by the Oracle Access Manager 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 15–5 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_Component_Name*

    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.

15.3.5 Configuring IP Validation for the Enterprise Deployment Webgate

IP validation determines if a client's IP address is the same as the IP address stored in the ObSSOCookie cookie generated for single sign-on. IP validation can cause issues in systems using load balancer devices configured to perform IP termination or when the authenticating webgate is front-ended by a different load balancing router (LBR) or proxy than the one front-ending the enterprise deployment.

To make sure your enterprise deployment's LBR or proxy are not validated in these cases:

1. Open the Access System Console and log in as an administrator at the following URL:

   `http://`*`host_name`*`:`*`port`*`/access/oblix`

   where *`host_name`* refers to the host where the 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.
3. On the Access System Console main page, click **Access System Configuration**, and then click the **Access Gate Configuration** link in the left pane to display the AccessGates Search page.
4. Enter the appropriate search criteria and click **Go** to display a list of access gates.
5. Select the access gate created by the OAM Configuration Tool.
6. Click **Modify** at the bottom of the page.
7. In the IPValidationException field, enter the IP address of the load balancer or proxy front-ending the enterprise deployment.
8. Click **Save** at the bottom of the page.

15.3.6 Setting up the Oracle Access Manager Identity Asserter

This section is based on the assumption that you have already set up the LDAP authenticator by following the steps in Section 15.2.2.1, "Creating the LDAP Authenticator." If you have not already created the LDAP authenticator, do it before continuing with this section.

This section covers the following topics:

- Section 15.3.6.1, "Back Up Configuration Files"
- Section 15.3.6.2, "Setting Up the Oracle Access Manager Identity Asserter"
- Section 15.3.6.3, "Setting the Order of Providers"

15.3.6.1 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/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.

15.3.6.2 Setting Up the Oracle Access Manager Identity Asserter

To set up the Oracle Access Manager Identity Asserter:

1. Log in to the WebLogic Server Administration Console, if not already logged in.
2. Click **Lock & Edit**.
3. Navigate to **SecurityRealms**, then the default realm name, and then **Providers**.
4. Click **New** and select **OAMIdentityAsserter** from the dropdown menu.
5. Name the asserter (for example, `OAM ID Asserter`) and click **OK**.
6. Click the newly added asserter to see the configuration screen for the Oracle Access Manager Identity Asserter.
7. Set the control flag to `REQUIRED` and click **Save**.
8. Open the **Provider Specific** tab to configure the following required settings:
   - **Primary Access Server:** provide Oracle Access Manager server endpoint information in *host*:*port* format.
   - **AccessGate Name:** name of the AccessGate (for example, `ECM_EDG_AG`).
   - **AccessGate Password:** password for the AccessGate (optional).
9. Save the settings.

15.3.6.3 Setting the Order of Providers

Reorder the Oracle Access Manager Identity Asserter, the Oracle Internet Directory or Oracle Virtual Directory Authenticator, and the 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`
- DefaultIdentityAsserter

After reordering, save the settings, activate the changes, and restart all servers.

> **Note:** Do not forget to create a new credential for the new user. See Section 11.18, "Configuring BPEL CSF Credentials" for further details. (This book uses the `weblogic_ecm` user as an example for SSO.)

15.4 Oracle Access Manager 11g Integration

This section describes how to set up Oracle Access Manager 11*g* as the single sign-on solution for the Oracle WebCenter Content enterprise deployment topology. It contains the following sections:

- Section 15.4.1, "Overview of Oracle Access Manager Integration"
- Section 15.4.2, "Prerequisites for Oracle Access Manager"
- Section 15.4.3, "Setting Up WebGate"
- Section 15.4.4, "Registering the WebGate Agent"
- Section 15.4.5, "Setting Up the WebLogic Server Authenticators"

15.4.1 Overview of Oracle Access Manager Integration

Oracle Access Manager is the recommended single sign-on solution for Oracle Fusion Middleware 11*g* Release 1. For more information on installing and configuring an Oracle Access Manager installation, see the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*. This section explains the procedure for configuring the Oracle WebCenter Content installation with an existing Oracle Access Manager 11*g* 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 Oracle WebCenter Content enterprise deployment topology described in this guide uses a single sign-on configuration where both the Oracle WebCenter Content system and the single sign-on system are in the same network domain (`mycompany.com`). For a multidomain configuration, see the required configuration steps in "Introduction to the OAM Policy Model, Single Sign-On" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager with Oracle Security Token Service*.

15.4.2 Prerequisites for Oracle Access Manager

The setup for Oracle Access Manager assumes an existing Oracle Access Manager 11*g* installation complete with Access Managers and a policy protecting the Policy Manager. For more information on installing and configuring an Oracle Access Manager installation, see *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*. This setup includes a directory service such as Oracle Internet Directory, either standalone or as part of an Oracle Virtual Directory configuration.

In addition, the Oracle Access Manager installation should have its own Web server configured with WebGate. This section also provides the steps for using the Oracle Access Manager Web server as a delegated authentication server.

15.4.3 Setting Up WebGate

You must set up a WebGate on each of the WEBHOST machines where Oracle HTTP Server has already been installed.

This section covers the following topics:

- Section 15.4.3.1, "Installing GCC Libraries"
- Section 15.4.3.2, "Installing WebGate"
- Section 15.4.3.3, "Post-Installation Steps"

15.4.3.1 Installing GCC Libraries

You must download and install third-party GCC libraries on each of the WEBHOST machines before installing WebGate. You can download the appropriate GCC library from the following third-party website:

`http://gcc.gnu.org`

For 32-bit Linux, the required libraries are libgcc_s.so.1 and libstdc++.so.5 with a version number of 3.3.2. Table 15–3 lists the versions of third-party GCC libraries for Linux and Solaris.

Table 15–3 Versions of GCC Third-Party Libraries for Linux and Solaris

Operating System	Architecture	GCC Libraries	Required Library Version
Linux 32-bit	x86	libgcc_s.so.1 libstdc++.so.5	3.3.2
Linux 64-bit	x64	libgcc_s.so.1 libstdc++.so.6	3.4.6
Solaris 64-bit	SPARC	libgcc_s.so.1 libstdc++.so.5	3.3.2

15.4.3.2 Installing WebGate

This section describes the procedures for installing WebGate. You must install WebGate on each of the WEBHOST machines.

Launching the Installer

The installer program for Oracle HTTP Server 11*g* Webgate for Oracle Access Manager is included in the webgate.zip file.

To start the installation wizard:

1. Extract the contents of the webgate.zip file to a directory. By default, this directory is named webgate.
2. Move to the Disk1 subdirectory under the webgate directory.

3. Set the *WEB_HOME* environment variable to the Middleware home for the web tier:

```
export MW_HOME=ORACLE_BASE/product/fmw
export WEB_HOME=MW_HOME/web
```

4. Start the installer using the following command:

```
$ ./runInstaller -jreLoc MW_HOME/jdk
```

> **Note:** When you install Oracle HTTP Server, the `jdk` directory is created under the Middleware home directory. You must enter the absolute path of the JRE folder located in this JDK when launching the installer.

After the installer starts, the Welcome screen opens.

Installation Flow and Procedure

If you need additional help with any of the installation screens, click **Help** to access the online help.

To install Oracle HTTP Server 11*g* Webgate for Oracle Access Manager:

1. In the Welcome screen, click **Next**.
2. In the Prerequisite Checks screen, click **Next**.
3. In the Specify Installation Location screen, specify the **Oracle Middleware Home** and **Oracle Home Directory** locations:
   - *ORACLE_BASE*`/product/fmw`
   - `Oracle_OAMWebGate1` (leave the default name)

   > **Note:** The Middleware home contains an Oracle home for Oracle Web Tier. The default name is `Oracle_OAMWebGate1` for this Oracle home directory, which will be created under the Middleware home.

   Click **Next**.
4. In the Specify GCC Library screen, specify the directory that contains the GCC libraries, or click **Browse** to navigate to their location on your local computer (see Section 15.4.3.1, "Installing GCC Libraries"), and click **Next**.
5. In the Installation Summary screen, verify the information on this screen and click **Install** to begin the installation.
6. In the Installation Progress screen, you may be prompted to run the *ORACLE_HOME*`/oracleRoot.sh` script to set up the proper file and directory permissions.

   Click **Next** to continue.
7. In the Installation Complete screen, click **Finish** to exit the installer.

15.4.3.3 Post-Installation Steps

Complete the following procedure on each of the WEBHOST machines after installing Oracle HTTP Server 11*g* Webgate for Oracle Access Manager:

1. Move to the following directory under your Oracle Home for Webgate:

   ```
   $ cd Webgate_Oracle_Home/webgate/ohs/tools/deployWebGate
   ```

   Webgate_Oracle_Home is the directory where you have installed Oracle HTTP Server Webgate and created the Oracle Home for Webgate; for example:

   ```
   MW_HOME/web/Oracle_OAMWebGate1
   ```

   Note: Oracle_OAMWebGate1 is the default.

2. On the command line, run the following command (on a single line) to copy the required bits of agent from the `Webgate_Oracle_Home` directory to the Webgate Instance location:

   ```
   $ ./deployWebGateInstance.sh -w ORACLE_BASE/admin/webN/config/OHS/ohsN
   -oh Webgate_Oracle_Home
   ```

   The *ORACLE_BASE*`/admin/web`*N*`/config/OHS/ohs`*N* directory is the Instance Home of an Oracle HTTP Server (where *N* is a sequential number for your installation; for example, `1` for `WEBHOST1` or `2` for `WEBHOST2`).

   Note: An instance home for Oracle HTTP Server is created after you configure Oracle HTTP Server.

3. Run the following command to ensure that the *LD_LIBRARY_PATH* variable contains *Oracle_Home_for_Oracle_HTTP_Server*`/lib`:

   ```
   $ export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:ORACLE_BASE/product/fmw/webN/lib
   ```

4. From your present working directory, move up one directory level:

   ```
   $ cd Webgate_Oracle_Home/webgate/ohs/tools/setup/InstallTools
   ```

5. On the command line, run the following command (on a single line) to copy apache_webgate.template from the *Webgate_Oracle_Home* directory to the Webgate Instance location (renamed to webgate.conf) and update the httpd.conf file to add one line to include the name of webgate.conf:

   ```
   $ ./EditHttpConf -w ORACLE_BASE/admin/webN/config/OHS/ohsN [-oh Webgate_Oracle_
   Home]
   [-o output_file]
   ```

   Note: The `-oh` *WebGate_Oracle_Home* and `-o` *output_file* parameters are optional.

   In the command, *WebGate_Oracle_Home* is the directory where you have installed Oracle HTTP Server Webgate for Oracle Access Manager and created as the Oracle Home for Webgate, for example:

   ```
   MW_HOME/Oracle_OAMWebGate1
   ```

The *ORACLE_BASE*/admin/web*N*/config/OHS/ohs*N*/ directory is the instance home of Oracle HTTP Server, where *N* is a sequential number for your installation; for example, 1 for WEBHOST1 or 2 for WEBHOST2.

The *output_file* value is the name of the temporary output file used by the tool; for example:

```
Edithttpconf.log
```

15.4.4 Registering the WebGate Agent

This section describes the procedures for registering the WebGate Agent on each of the WEBHOST machines:

- Section 15.4.4.1, "Extracting and Using the RREG Tool"
- Section 15.4.4.2, "Updating the Oracle Access Manager 11g Request File"
- Section 15.4.4.3, "Running the oamreg Tool"
- Section 15.4.4.4, "Changing the inspection.wsil Resource to Use the Basic Authentication Scheme"
- Section 15.4.4.5, "Updating the Oracle Access Manager 11g Server Configuration to Support the Basic Cookieless Scheme"
- Section 15.4.4.6, "Copying Access Files to WEBHOST Machines"

15.4.4.1 Extracting and Using the RREG Tool

The RREG tool is part of the Oracle Access Manager 11*g* installation. If it is not already available, extract it on each of the Oracle Identity Management machines with the following procedure:

1. After installing and configuring Oracle Access Manager, navigate to the following location:

   ```
   IDM_Home/oam/server/rreg/client
   ```

2. On the command line, untar the RREG.tar.gz file using gunzip, as in the following example:

   ```
   gunzip RREG.tar.gz

   tar -xzvf RREG.tar
   ```

3. Edit the oamreg.sh script in the *RREG_HOME*/bin/ directory and change the OAM_REG_HOME parameter according to your setup:

   ```
   OAM_REG_HOME=RREG_Home
   ```

   (where *RREG_Home* is the directory to which you extracted the contents of RREG.tar.gz and rreg).

   Save the script file.

The RREG Configuration Tool provides a way to register protected and public resources into the Oracle Access Manager system. The list of protected resources to be added to the Oracle Access Manager system is as follows:

```
/adfAuthentication
/imaging/faces
/em
/console
```

```
/DefaultToDoTaskFlow
/sdpmessaging/userprefs-ui
/integration/worklistapp
/workflow/sdpmessagingsca-ui-worklist
/soa/composer
/soa-infra
/soa-infra/deployer
/soa-infra/events/edn-db-log
/soa-infra/cluster/info
/inspection.wsil
```

The list of public resources follows:

```
/cs
/_ocsh
/imaging
/soa-infra/directWSDL
```

The list of excluded resources follows:

```
/wsm-pm
/soa-infra/services
/ucs/messaging/webservice
```

15.4.4.2 Updating the Oracle Access Manager 11g Request File

The *RREG_Home*/input/ directory contains a template file named OAM11gRequest.xml. Copy this file to WCCOAM11gRequest.xml, and edit that file to create the policies for the Oracle WebCenter Content installation.

> **Note:** Replace *OAM_HOST*, *OAM_ADMINSERVER_PORT*, *WCC_EDG_AGENT*, and *WCC_EDG_DOMAIN* with their respective values in your installation.

After editing, the file should look as follows:

```
<?xml version="1.0" encoding="UTF-8"?>

<OAM11GRegRequest>
    <serverAddress>http://OAM_HOST:OAM_ADMINSERVER_PORT</serverAddress>
    <agentName>WCC_EDG_AGENT</agentName>
    <applicationDomain>WCC_EDG_DOMAIN</applicationDomain>
    <cachePragmaHeader>private</cachePragmaHeader>
    <cacheControlHeader>private</cacheControlHeader>
    <ipValidation>1</ipValidation>
    <ValList ListName="ipValidationExceptions">
        <ValListMember Value="10.1.1.1"/>
    </ValList>
    <logOutUrls>
        <url>/oamsso/logout.html</url>
    </logOutUrls>

    <protectedResourcesList>
        <resource>/adfAuthentication</resource>
        <resource>/imaging/faces</resource>
        <resource>/em</resource>
        <resource>/console</resource>
        <resource>/DefaultToDoTaskFlow</resource>
        <resource>/sdpmessaging/userprefs-ui</resource>
```

```
        <resource>/integration/worklistapp</resource>
        <resource>/workflow/sdpmessagingsca-ui-worklist</resource>
        <resource>/soa/composer</resource>
        <resource>/soa-infra/deployer</resource>
        <resource>/soa-infra/events/edn-db-log</resource>
        <resource>/soa-infra/cluster/info</resource>
        <resource>/inspection.wsil</resource>
        <resource>/soa-infra</resource>
     </protectedResourcesList>

     <publicResourcesList>
        <resource>/cs</resource>
        <resource>/_ocsh</resource>
        <resource>/imaging</resource>
        <resource>/soa-infra/directWSDL</resource>

   </publicResourcesList>
   <excludedResourcesList>
        <resource>/wsm-pm</resource>
        <resource>/soa-infra/services</resource>
        <resource>/ucs/messaging/webservice</resource>
   </excludedResourcesList>
   <userDefinedParameters>

        <userDefinedParam>
            <name>filterOAMAuthnCookie</name>
            <value>false</value>
        </userDefinedParam>

   </userDefinedParameters>

</OAM11GRegRequest>
```

Notes:

- The *resource_name*`/.../*` resources will be automatically added during the registration.
- This guide describes the validation field entry in request files for Oracle Access Manager 11*g* (11.1.1.2) and later. The validation exception list is defined differently in earlier versions of Oracle Access Manager 11*g*. For earlier versions, instead of using the `<ValList>` entry as shown in the preceding text, use this syntax after the `</publicResourcesList>` entry:

  ```
  <userDefinedParameters>
      <userDefinedParam>
          <name>ipValidationExceptions</name>
          <value>10.1.1.1</value>
      </userDefinedParam>
  </userDefinedParameters>
  ```

  For more information about adding IP validation exceptions, see the *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager with Oracle Security Token Service*.

15.4.4.3 Running the oamreg Tool

Run the oamreg tool with the following command:

```
$ ./RREG_Home/bin/oamreg.sh inband input/WCCOAM11gRequest.xml
```

The run should look as follows:

```
-----------------------------------------------
Welcome to OAM Remote Registration Tool!
Parameters passed to the registration tool are:
Mode: inband
Filename: /u01/app/oracle/product/fmw/iam/oam/server/rreg/client/rreg/
input/WCCOAM11GRequest.xml
Enter admin username:oamadmin
Username: your_oamadmin_user
Enter admin password: your_oamadmin_password
Do you want to enter a Webgate password?(y/n): y
Enter webgate password: your_webgate_password
Enter webgate password again: your_webgate_password
Password accepted. Proceeding to register..
Apr 18, 2011 12:22:36 PM
oracle.security.am.engines.rreg.client.handlers.request.OAM11GRequestHandler
getWebgatePassword
INFO: Passwords matched and accepted.
Do you want to import an URIs file?(y/n): n

---------------------------------------
Request summary:
OAM11G Agent Name:WCC_EDG_AGENT
URL String:WCC_EDG_AGENT
Registering in Mode:inband
Your registration request is being sent to the Admin server at:
http://oamserver.mycompany.com:7001
---------------------------------------

Inband registration process completed successfully! Output artifacts are created
in the output folder.
```

15.4.4.4 Changing the inspection.wsil Resource to Use the Basic Authentication Scheme

By default, the `inspection.wsil` resource is set to use the form authentication scheme. For the connection between the workflow and Oracle WebCenter Content: Imaging to work, this resource must be updated on each of the WEBHOST machines to use the basic authentication scheme instead:

1. Log in to the Oracle Access Manager console at `http://`*`OAM_HOST`*`:`*`OAM_ADMINSERVER_PORT`*`/oamconsole`.
2. Using the navigation tree on the left, choose **Application Domains** and then the application domain name to navigate to the application domain created (*`WCC_EDG_DOMAIN`*).
3. Expand your application domain's name.
4. Expand **Authentication Policies**.
5. Double-click **Protected Resource Policy**.
6. Select the `inspection.wsil` and `inspection.wsil/.../*` resources, and click the **Delete** icon in the Resources pane to remove them.

7. Click **Apply**, and confirm the action when prompted.
8. In the navigation tree, click **Authentication Policies** again, and click the **Create** button in the tool bar above the navigation tree.
   a. Enter a name for the policy (for example, `New Basic Policy`).
   b. Select **BasicSessionlessScheme** as the authentication scheme.
   c. Click **Apply**.

      You will see the newly created policy under **Authentication Policies** in the navigation tree.
   d. Open the newly created policy.
   e. On the Resources pane, click the add icon (plus sign) on the right, and add the `inspection.wsil` and `inspection.wsil/.../*` resources.
   f. Click **Apply**.
9. Click the refresh icon on the navigation tree, and verify the new authentication policy (click it, and make sure the `inspection.wsil` and `inspection.wsil/.../*` resources were added).

Note: Do not forget to create a new credential for the new user. See Section 11.18, "Configuring BPEL CSF Credentials" for further details. (This book uses the `weblogic_ecm` user as an example for SSO.)

15.4.4.5 Updating the Oracle Access Manager 11g Server Configuration to Support the Basic Cookieless Scheme

You must set the `NoUniqueSessionsFor10gAgents` parameter in the Oracle Access Manager 11*g* configuration to `true` on each of the WEBHOST machines. To do this, edit the `oam-config.xml` file located in the *`IDM_Home`*`/oam/server/config/` directory, and change the line

```
<Setting Name="NoUniqueSessionsFor10gAgents" Type="xsd:string">false</Setting>
```

to

```
<Setting Name="NoUniqueSessionsFor10gAgents" Type="xsd:string">true</Setting>
```

Save the file, and restart the Oracle Access Manager server in your Oracle Identity Management system for the change to take effect.

15.4.4.6 Copying Access Files to WEBHOST Machines

The following two files are generated in *`RREG_Home`*`/output/`*`WCC_EDG_AGENT`*:

- `ObAccessClient.xml`
- `cwallet.sso`

Copy these files to the WebGate instance location on each of the WEBHOST machines:

```
scp ObAccessClient.xml oracle@WEBHOSTN:ORACLE_BASE/admin/webN/config/OHS/ohsN/
webgate/config/

scp cwallet.sso oracle@WEBHOSTN:ORACLE_BASE/admin/webN/config/OHS/ohsN/
webgate/config/
```

In the `scp` command, *`N`* is a sequential number for your installation; for example, `1` for `WEBHOST1` or `2` for `WEBHOST2`.

15.4.5 Setting Up the WebLogic Server Authenticators

This section is based on the assumption that you have already set up the LDAP authenticator by following the steps in Section 15.2.2.1, "Creating the LDAP Authenticator." If you have not already created the LDAP authenticator, do it before continuing with this section.

This section covers the following topics:

- Section 15.4.5.1, "Backing Up Configuration Files"
- Section 15.4.5.2, "Setting Up the Oracle Access Manager Identity Asserter"
- Section 15.4.5.3, "Setting the Order of Providers"

15.4.5.1 Backing 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-config.xml
ORACLE_BASE/admin/domain_name/aserver/domain_
name/config/fwmconfig/system-jazn-data.xml
```

In addition, back up the boot.properties file for the Administration Server.

15.4.5.2 Setting Up the Oracle Access Manager Identity Asserter

To set up the Oracle Access Manager Identity Asserter:

1. Log in to the WebLogic Server Administration Console, if not already logged in.
2. Click **Lock & Edit.**
3. Navigate to **SecurityRealms**, then the default realm name, and then **Providers**.
4. Click **New**, and select the asserter type **OAMIdentityAsserter** from the dropdown menu.
5. Name the asserter (for example, `OAM ID Asserter`) and click **OK**.
6. Click the newly added asserter to see the configuration screen for the Oracle Access Manager Identity Asserter.
7. Set the control flag to `REQUIRED`.
8. Select both the **ObSSOCookie** and **OAM_REMOTE_USER** options under Chosen types.
9. Save the settings, and click **Activate Changes** to propagate the changes.

Finally, log in to the WLST console as an administrator, and run the following command:

```
addOAMSSOProvider(loginuri="/${app.context}/adfAuthentication",logouturi="/oamsso/
logout.html")
```

15.4.5.3 Setting the Order of Providers

To set the order of the providers:

1. Log in to the WebLogic Server Administration Console, if not already logged in.
2. Click **Lock & Edit.**
3. Navigate to **SecurityRealms**, then the default realm name, and then **Providers**.
4. Reorder the Oracle Access Manager Identity Asserter, the Oracle Internet Directory or Oracle Virtual Directory Authenticator, and the 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
5. Click **OK**.
6. Click **Activate Changes** to propagate the changes.
7. Restart the Administration Server and all Managed Servers.

15.5 Validating Access Through the Load Balancer and SSO

Validate single sign-on through the front end (using the SSO user name and password):

- `http://admin.mycompany.com/console`
- `http://admin.mycompany.com/em`
- `http://wcc.mycompany.com/cs`
- `http://wcc.mycompany.com/imaging`

15.6 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 that 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, see the "Backup and Recovery Recommendations for Oracle HTTP Server" section in this guide. For information on how to recover components, see the "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" section in the guide. For information about database backup, see the *Oracle Database Backup and Recovery User's Guide*.

To back up the installation at this point:

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 on SOAHOST1. Perform a backup to save your domain configuration. The configuration files all exist under the *ORACLE_BASE*/ admin/*domain_name*/ directory.

   ```
   tar -cvpf edgdomainback.tar ORACLE_BASE/admin/domain_name
   ```

16

Managing the Topology

This chapter describes some operations that you can perform after you have set up the topology, including monitoring, scaling, and backing up your topology. It contains the following sections:

16.1 Overview of Managing the Oracle WebCenter Content Topology

After configuring the Oracle WebCenter Content enterprise deployment, you can use the information in this chapter to manage the topology. Table 16–1 lists some tasks you can perform to manage the topology.

Table 16–1 Tasks for Managing the Oracle WebCenter Content Topology

Task	Description	More Information
Configure Imaging for high performance, scalability, and high availability in an Imaging cluster	Define an optimal input file strategy	Section 16.2, "Defining an Optimal Input File Strategy for Oracle WebCenter Content: Imaging"
Manage the Oracle SOA Suite subsystem used by Imaging	Deploy SOA composites to a server address, manage space in the Oracle SOA Suite infrastructure, and configure UMS drivers.	Section 16.3, "Deploying Composites and Artifacts in the Oracle WebCenter Content Enterprise Deployment Topology" Section 16.4, "Managing Space in the SOA Infrastructure Database" Section 16.5, "Configuring UMS Drivers for Oracle WebCenter Content: Imaging"
Expand the topology by scaling it up, or out	Add new Managed Servers to nodes, or add new nodes.	Section 16.6, "Scaling Up the Oracle WebCenter Content Topology" Section 16.7, "Scaling Out the Oracle WebCenter Content Topology"
Verify that manual failover of the Administration Server works correctly	Fail over the Administration Server to another node.	Section 16.8, "Verifying Manual Failover of the Administration Server"
Back up the topology before and after any configuration changes	Back up directories and files to protect against failure as a result of configuration changes.	Section 16.9, "Performing Backups and Recoveries in Oracle WebCenter Content Enterprise Deployments"
Prevent connection timeouts	Configure the firewall so that the database connection is not timed out.	Section 16.10, "Preventing Timeouts for SQLNet Connections"
Configure Oracle Web Services Manager (Oracle WSM) security policies for Oracle WebCenter Content and Imaging web services	Use the appropriate Oracle WSM policy enforcements instead of basic HTTP authentication.	Section 16.11, "Configuring Oracle Web Service Manager Security Policies for Oracle WebCenter Content and Imaging Services"
Troubleshoot problems	Implement solutions for known issues that may occur after you have configured the topology.	Section 16.12, "Troubleshooting the Oracle WebCenter Content Enterprise Deployment Topology"

For more information about managing Oracle WebCenter Content, see the following documents:

- *Oracle® Fusion Middleware Administering Oracle WebCenter Content*
- *Oracle Fusion Middleware Managing Oracle WebCenter Content*
- *Oracle WebCenter Content Administrator's Guide for Imaging*

16.2 Defining an Optimal Input File Strategy for Oracle WebCenter Content: Imaging

The input file is the smallest unit of work that the input agent can schedule and process. There are multiple elements to be taken into consideration to achieve the highest performance, scalability, and high availability in an Imaging cluster:

- All of the machines in an Imaging cluster share a common input directory.
- Input files from this directory are distributed to each machine through a JMS queue.

- The frequency with which the directory is polled for new files is configurable.
- Each machine has multiple parsing agents that process the input files. The number of parsing agents is configured through the Work Manager created within the Oracle WebCenter Content: Imaging deployment.

Optimum performance will be achieved when:

- Each Imaging cluster instance has the maximum affordable number of parsing agents configured through the Work Manager without compromising the performance of the other Imaging activities, such as the user interface and Web services.
- The inbound flow of documents is partitioned into input files containing the appropriate number of documents. On average there should be two input files queued for every parsing agent within the cluster.
- If one or more machines within a cluster fails, active machines will continue processing the input files. Input files from a failed machine will remain in limbo until the server is restarted. Smaller input files ensure that machine failures do not place large numbers of documents into this limbo state.

For example, consider 10,000 inbound documents per hour being processed by two servers. A configuration of two parsing agents per server produces acceptable overall performance and ingests two documents per second per agent. The four parsing agents at two documents per second is eight documents per second, or 28,800 documents per hour. Note that a single input file of 10,000 documents will not be processed in an hour since a single parsing agent working at 7,200 documents per hour will be unable to complete it. However, if you divide the single input file up into eight input files of 1,250 documents, this ensures that all four parsing agents are fully utilized, and the 10,000 documents are completed in the one hour period. Also, if a failure should occur in one of the servers, the other can continue processing the work remaining on its parsing agents until the work is successfully completed.

16.3 Deploying Composites and Artifacts in the Oracle WebCenter Content Enterprise Deployment Topology

When deploying SOA composites to the Oracle SOA Suite subsystem used by Oracle WebCenter Content: Imaging, deploy to a specific server's address and not to the load balancer address (wcc.mycompany.com). Deploying to the load balancer address may require direct connection from the deployer nodes to the external load balancer address, which may require additional ports to be opened in the firewalls used by the system.

16.4 Managing Space in the SOA Infrastructure Database

Although not all composites may use the database frequently, the service engines generate a considerable amount of data in the CUBE_INSTANCE and MEDIATOR_INSTANCE schemas. Lack of space in the database may prevent SOA composites from functioning. Watch for generic errors, such as `oracle.fabric.common.FabricInvocationException`, in Oracle Enterprise Manager Fusion Middleware Control (dashboard for instances). Search also in the Oracle SOA Suite server's logs for errors, such as:

```
Error Code: 1691
...
ORA-01691: unable to extend lob segment
SOAINFRA.SYS_LOB0000108469C00017$$ by 128 in tablespace SOAINFRA
```

These messages are typically indicators of space issues in the database that may likely require adding more data files or more space to the existing files. The SOA database administrator should determine the extension policy and parameters to be used when adding space. Additionally, old composite instances can be purged to reduce the SOA infrastructure database size. Oracle does not recommend using Oracle Enterprise Manager Fusion Middleware Control for this type of operation because in most cases such operations cause a transaction timeout.

Refer to "Managing SOA Composite Applications" in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite* for more details on the possible operations included in the SQL packages provided. Always use the scripts provided for a correct purge. Deleting rows in just the composite_dn table may leave dangling references in other tables used by the Oracle Fusion Middleware SOA Infrastructure.

16.5 Configuring UMS Drivers for Oracle WebCenter Content: Imaging

Note: This step is required only if the Oracle SOA Suite system used by Oracle WebCenter Content: Imaging is using Unified Messaging System (UMS).

UMS driver configuration is not automatically propagated in an Oracle SOA Suite cluster. When UMS is used by the Oracle SOA Suite system that Oracle WebCenter Content: Imaging invokes, this implies that you need to do the following:

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/mserver/domain_name/servers/server_name/tmp/
   _WL_user/ums_driver_name/*/configuration/driverconfig.xml
   ```

   (where * represents a directory whose name is randomly generated by Oracle WebLogic Server during deployment; for example, `3682yq`).

To create the file in preparation for possible failovers, users can force a server migration and copy the file from the source node. For example:

1. Configure the driver for WLS_IMG1 in SOAHOST1.
2. Force a failover of WLS_IMG1 to SOAHOST2. Verify the directory structure for the UMS driver configuration in the failover node:

   ```
   cd ORACLE_BASE/admin/domain_name/mserver/domain_name/servers/server_name/tmp/_
   WL_user/ums_driver_name/*/configuration
   ```

   (where * represents a directory whose name is randomly generated by WLS during deployment, for example, `3682yq`).

3. Run the following command on SOAHOST1 to do a remote copy of the driver configuration file from SOAHOST1 to SOAHOST2:

   ```
   scp ORACLE_BASE/admin/domain_name/mserver/domain_name/servers/server_name/tmp/_
   WL_user/ums_driver_name/*/configuration/driverconfig.xml
   oracle@SOAHOST2:ORACLE_BASE/admin/domain_name/mserver/domain_name/servers/
   server_name/tmp/_WL_user/ums_driver_name/*/configuration/
   ```

   (where * represents a directory whose name is randomly generated by Oracle WebLogic Server during deployment, for example; `3682yq`).

It is required to restart the driver for these changes to take effect (that is, for the driver to consume the modified configuration):

1. Log in to the Oracle WebLogic Server Administration Console.
2. Expand the environment node on the navigation tree.
3. Click **Deployments**.
4. Select the driver.
5. Click **Stop** and then **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** and then **Servicing all requests** and confirm the operation.

Make sure that you verify in Fusion Middleware Control that the properties for the driver have been preserved.

16.6 Scaling Up the Oracle WebCenter Content Topology

When you scale up the topology, you add new Managed Servers to nodes that are already running on one or more Managed Servers. You already have a node that runs a Managed Server that is configured with the necessary components. The node contains a WebLogic Server home and a Middleware home in shared storage. Use these existing installations (such as WebLogic Server home, Middleware home, and domain directories) when you create the new Managed Servers. You do not need to install WebLogic Server binaries at a new location or to run `pack` and `unpack`.

> **Note:** A shared domain directory for a Managed Server with Content Server does not work because certain files within the domain, such as `intradoc.cfg`, are specific to each node. To prevent issues with node-specific files, use a local (per node) domain directory for each Oracle WebCenter Content and Oracle WebCenter Content: Inbound Refinery Managed Server.

The scale-up procedure depends on the topology component:

- Section 16.6.1, "Scale-Up Procedure for Oracle WebCenter Content"
- Section 16.6.2, "Scale-Up Procedure for Oracle WebCenter Content: Imaging"
- Section 16.6.3, "Scale-Up Procedure for Oracle SOA Suite"

16.6.1 Scale-Up Procedure for Oracle WebCenter Content

Only one Oracle WebCenter Content Managed Server per node per domain is supported by Oracle Fusion Middleware. To add additional Oracle WebCenter Content Managed Servers, follow the steps in Section 16.7.1, "Scale-Out Procedure for Oracle WebCenter Content," to add an Oracle WebCenter Content Managed Server to a new node.

16.6.2 Scale-Up Procedure for Oracle WebCenter Content: Imaging

Before scaling up the Imaging servers, you must enable a VIP on WCCHOST1 (say, WCCHOST1VHN2), and you must also correctly resolve the host names in the network system used by the topology (either by DNS server or host resolution). To enable the VIPs, follow the example described in Section 9.2, "Enabling SOAHOST1VHN1 on SOAHOST1 and SOAHOST2VHN1 on SOAHOST2."

To scale up the Imaging servers in the enterprise deployment topology:

1. Using the WebLogic Server Administration Console, clone WLS_IMG1 to 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:

   a. In the Domain Structure window of the WebLogic Server Administration Console, expand the **Environment** node and then **Servers**.

   b. On the Summary of Servers page, click **Lock & Edit** and then select the Managed Server that you want to clone (WLS_IMG1).

   c. Click **Clone**.

   d. Name the new Managed Server WLS_IMG*n*, where *n* is a number that identifies the new Managed Server.

   e. For the server listen address, assign the host name or IP to use for this new Managed Server. If you are planning to use server migration for this server (which Oracle recommends), this should be the virtual host name for the server. This virtual host name should be different from the one used for the existing Managed Server.

   f. For the server listen port, enter the listening port number for the Imaging cluster (IMG_Cluster). The reference topology in this book uses port number 16000.

   g. Click **OK** and **Activate Changes**. You should now see the newly created server WLS_IMG*n* in the summary of servers.

   The remainder of the steps that follow are based on the assumption that you are adding a new server to WCCHOST1, which is already running WLS_IMG1.

2. Configure a JMS persistence store and JMS servers for Oracle WebCenter Content: Imaging JMS.

   Configure the location for the JMS persistence stores as a directory that is visible from both nodes. By default, the JMS servers used by Oracle WebCenter Content: Imaging are configured with no persistent store and use the WebLogic Server store (*ORACLE_BASE*/admin/*domain_name*/mserver/*domain_name*/servers/*server_name*/data/store/default).

You must change Oracle JMS server persistent store to use a shared base directory, as follows:

a. Log in to the WebLogic Server Administration Console.

b. In the Domain Structure window, expand the **Services** node and then click the **Persistence Stores** node.

c. On the Summary of Persistence Stores page, click **Lock & Edit**.

d. Click **New**, and then **Create FileStore**.

e. On the Create a New File Store page, enter the following information:

- **Name:** IMGJMSServer*n*Store (for example, IMGJMSServer3Store, which allows you identify the service it is created for)

- **Target:** WLS_IMG*n* (for example, WLS_IMG3).

- **Directory:** Specify a directory that is located in shared storage so that it is accessible from both WCCHOST1 and WCCHOST2 (*ORACLE_BASE*/admin/*domain_name*/*img_cluster_name*/jms).

Note: This directory must exist before the Managed Server is started or the start operation will fail.

---

f. Click **OK** and activate the changes.

g. In the Domain Structure window, expand the **Services** node, then the **Messages** node, and then click **JMS Servers**.

h. On the Summary of JMS Servers page, click **New** and then enter the following information:

- **Name:** IpmJmsServer*n* (for example, IpmJmsServer3)

- **Persistent Store:** select the persistence store that you created above: IMGJMSServer*n*Store (for example, IMGJMSServer3Store).

Click **Next** and then specify WLS_IMG*n* (for example, WLS_IMG3) as the target. Click **Finish**.

i. Click **Save**, and activate the changes.

3. Configure a default persistence store for WLS_IMG*n* for transaction recovery:

a. Log in to the WebLogic Server Administration Console.

b. In the Domain Structure window, expand the **Environment** node and then click the **Servers** node.

c. On the Summary of Servers page, click WLS_IMG*n* (represented as a hyperlink) in the Name column of the table. The settings page for the WLS_IMG*n* server opens with the **Configuration** tab active.

d. Open the **Services** tab.

e. Click **Lock & Edit**.

f. 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*/*img_cluster_name*/tlogs

Note: This directory must exist before the Managed Server is started or the start operation will fail.

g. Click **Save** and activate the changes.

4. Disable host name verification for the new Managed Server, as described in Section 8.4.5, "Disabling Host Name Verification.".

   Before you can start and verify the WLS_IMG*n* Managed Server, you must disable host name verification. You can re-enable it after you have configured server certificates for the communication between the Administration Server and Node Manager in WCCHOST*n*. If the source server from which the new one has been cloned had already disabled host name verification, these steps are not required (the host name verification settings is propagated to the cloned server).

5. Start the newly created Managed Server (WLS_IMG):

   a. Log in to the WebLogic Server Administration Console.

   b. In the Domain Structure window, expand the **Environment** node and then click **Servers**.

   c. On the Summary of Servers page, open the **Control** tab, and shut down all existing WLS_IMG*n* Managed Servers in the cluster.

   d. Ensure that the newly created Managed Server, WLS_IMG*n*, is running.

6. Add the host name of the WLS_IMG*n* Managed Server (WCCHOST*n*VHN2) to the `SocketHostNameSecurityFilter` parameter list:

   a. Open the file *ORACLE_BASE*/admin/*domain_name*/*wcc_cluster_name*/cs/config/config.cfg in a text editor.

   b. Add the WLS_IMG*n* Managed Server listen addresses to the list of addresses that are allowed to connect to Oracle WebCenter Content: `SocketHostNameSecurityFilter=localhost|localhost.mycompany.com|WCCHOST1|WCCHOST2|WCCHOST`*n*`VHN`*n*

   c. Save the modified `config.cfg` file and restart the Oracle WebCenter Content servers for the changes to take effect.

7. Verify that server migration is configured for the new Managed Server.

   Note: Since this is a scale-up operation, the node should already contain a Node Manager and environment configured for server migration, so the following steps are only for verification. The floating IP for the new Imaging Managed Server should also be already present.

To verify that server migration is configured:

a. Log in to the WebLogic Server Administration Console.

b. In the Domain Structure window, expand the **Environment** node and then click **Servers**.

c. On the Summary of Servers page, click the name of the new Managed Server (represented as a hyperlink) in Name column of the table for which you want to configure migration.

d. On the settings page for the selected server, open the Migration subtab.

e. In the Migration Configuration section, select the servers that participate in migration in the **Available** window and click the right arrow. Select the same migration targets as for the servers that already exist on the node.

For example, for new Managed Servers on WCCHOST1, which is already running WLS_IMG1, select WCCHOST2. For new Managed Servers on WCCHOST2, which is already running WLS_IMG2, select WCCHOST1.

Note: The appropriate resources must be available to run the Managed Servers concurrently during migration.

---

f. Verify that the **Automatic Server Migration Enabled** option is selected.

This option enables Node Manager to start a failed server on the target node automatically.

g. Click **Save**.

h. Restart the Administration Server, Managed Servers, and Node Manager.

8. Test server migration for the new server. To test migration, perform the following steps from the node where you added the new server:

- Abruptly stop the WLS_IMG*n* Managed Server. To do this, run `kill -9 pid` on the PID of the Managed Server. You can identify the PID of the node using the following command:

```
ps -ef | grep WLS_IMGn
```

- Watch the Node Manager Console for a message indicating that WLS_IMG*n*'s floating IP has been disabled.
- Wait for Node Manager to attempt a second restart of WLS_IMG*n*. Node Manager waits for a fence period of 30 seconds before trying this restart.
- Once Node Manager restarts the server, stop it again. Node Manager should log a message indicating that the server will not be restarted again locally.

Note: After a server is migrated, to fail it back to its original node or machine, stop the Managed Server from the WebLogic Administration Console and then start it again. The appropriate Node Manager will start the Managed Server on the machine to which it was originally assigned.

16.6.3 Scale-Up Procedure for Oracle SOA Suite

To scale up the Oracle SOA Suite servers in the enterprise deployment topology:

Note: To scale up the Oracle SOA Suite subsystem used by Oracle WebCenter Content: Imaging, refer to the Oracle SOA Suite enterprise deployment topology documentation.

---

1. Using the WebLogic Server Administration Console, clone WLS_SOA1 to 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:

   a. In the Domain Structure window of the WebLogic Server Administration Console, expand the **Environment** node and then **Servers**.

   b. On the Summary of Servers page, select the Managed Server that you want to clone (WLS_SOA1).

   c. Click **Clone**.

   d. Name the new Managed Server WLS_SOA*n*, where *n* is a number that identifies the new Managed Server.

   Note: The remainder of the steps assume that you are adding a new server to SOAHOST1, which is already running WLS_SOA1.

   ---

2. For the listen address, assign the host name or IP to use for this new Managed Server. If you are planning to use server migration for this server (which Oracle recommends), this should be the VIP (also called a floating IP) to enable it to move to another node. The VIP should be different from the one used by the Managed Server that is already running.

3. Create JMS servers for Oracle SOA Suite and UMS on the new Managed Server:

   a. Use the WebLogic Server Administration Console to create a new persistent store for the new SOAJMSServer and name it, for example, `SOAJMSFileStore_N`. Specify the path for the store. This should be a directory on shared storage, as recommended in Section 4.3, "About Recommended Locations for the Different Directories":

   `ORACLE_BASE/admin/domain_name/soa_cluster_name/jms/`

   Note: This directory must exist before the Managed Server is started or the start operation fails.

   ---

   b. Create a new JMS server for Oracle SOA Suite (for example, SOAJMSServer_N). Use the `SOAJMSFileStore_N` for this JMS server. Target the SOAJMSServer_N server to the recently created Managed Server (WLS_SOA*n*).

c. Create a new persistence store for the new UMSJMSServer (for example, `UMSJMSFileStore_N`). Specify the path for the store. This should be a directory on shared storage, as recommended in Section 4.3, "About Recommended Locations for the Different Directories":

ORACLE_BASE/admin/*domain_name*/*soa_cluster_name*/jms/

Note: This directory must exist before the Managed Server is started or the start operation fails. You can also assign SOAJMSFileStore_N as the store for the new UMS JMS servers. For the purpose of clarity and isolation, individual persistent stores are used in the following steps.

---

d. Create a new JMS server for UMS (for example, UMSJMSServer_N). Use `UMSJMSFileStore_N` for this JMS server. Target the UMSJMSServer_N server to the recently created Managed Server (WLS_SOA*n*).

e. Update the subdeployment targets for the SOA JMS Module to include the recently created SOA JMS server. To do this, expand the **Services** node in the **Domain Structure** tree on the left of the WebLogic Server Administration Console, and then expand the **Messaging** node. Click **JMS Modules**. The JMS Modules page appears. Click **SOAJMSModule** (represented as a hyperlink in the Names column of the table). The Settings page for SOAJMSModule appears. Open the **SubDeployments** tab. The subdeployment module for SOAJMS appears.

Note: This subdeployment module name is a random name in the form of `SOAJMSServer`*XXXXXX* resulting from the Configuration Wizard JMS configuration for the first two servers (WLS_SOA1 and WLS_SOA2).

Click the **SOAJMSServer*XXXXXX*** subdeployment. Add the new JMS server for Oracle SOA Suite called SOAJMSServer_N to this subdeployment. Click **Save**.

f. Update the subdeployment targets for the UMSJMSSystemResource to include the recently created UMS JMS server. To do this, expand the **Services** node in the **Domain Structure** tree on the left of the WebLogic Server Administration Console and then expand the **Messaging** node. Click **JMS Modules**. The JMS Modules page appears. Click **UMSJMSSystemResource** (represented as a hyperlink in the Names column of the table). The Settings page for UMSJMSSystemResource appears. Open the **SubDeployments** tab. The subdeployment module for UMSJMS appears.

Note: This subdeployment module name is a random name in the form of `UCMJMSServer`*XXXXXX* resulting from the Configuration Wizard JMS configuration for the first two servers (WLS_SOA1 and WLS_SOA2).

Click the **UMSJMSServer*XXXXXX*** subdeployment. Add the new JMS server for UMS called UMSJMSServer_N to this subdeployment. Click **Save**.

4. Configure Oracle Coherence for deploying composites for the new server as described in Section 9.4, "Configuring Oracle Coherence for Deploying Composites."

Note: Only the `localhost` field needs to be changed for the server. Replace the `localhost` value with the listen address of the new server added: `Dtangosol.coherence.localhost=SOAHOST1VHNn`.

5. Configure a TX persistent store for the new server. This should be a location visible from other nodes as indicated in the recommendations about shared storage (see Section 4.3, "About Recommended Locations for the Different Directories").

   From the Administration Console, select the server name (WLS_SOA*n*) in the **Services** tab. Under **Default Store**, in **Directory**, enter the path to the folder where you want the default persistent store to store its data files:

   `ORACLE_BASE/admin/`*`domain_name`*`/`*`soa_cluster_name`*`/tlogs`

6. Disable host name verification for the new Managed Server.

   Before you can start and verify the WLS_SOA*n* Managed Server, you must disable host name verification. You can re-enable it after you have configured server certificates for the communication between the Administration Server and Node Manager in SOAHOST*n*. If the source server from which the new one has been cloned had already disabled host name verification, this step is not required (the host name verification settings are propagated to the cloned server).

   For more information, see Section 8.4.5, "Disabling Host Name Verification."

7. Start and test the new Managed Server from the WebLogic Server Administration Console:

   a. Ensure that the newly created Managed Server, WLS_SOA*n*, is running.

   b. Access the application on the LBR (`http://soainternal.mycompany.com/soa-infra`). The application should be functional.

   Note: The Oracle HTTP Servers in the topology should round-robin requests to the new added server (a few requests, depending on the number of servers in the cluster, may be required to hit the new server). Its is not required to add all servers in a cluster to the WebLogicCluster directive in the Oracle HTTP Server `*_vh.conf` files. However routing to new servers in the cluster will take place only if at least one of the servers listed in the WebLogicCluster directive is running.

8. Configure server migration for the new Managed Server.

   Note: Since this is a scale-up operation, the node should already contain a Node Manager and environment configured for server migration. The floating IP for the new Oracle SOA Suite Managed Server should also be already present.

To configure server migration:

a. Log in to the WebLogic Server Administration Console.

b. In the Domain Structure window, expand the **Environment** node and then click **Servers**.

c. On the Summary of Servers page, click the name of the new Managed Server (represented as a hyperlink) in Name column of the table for which you want to configure migration.

d. On the settings page for the selected server, open the Migration subtab.

e. In the Migration Configuration section, select the servers that participate in migration in the **Available** window and click the right arrow. Select the same migration targets as for the servers that already exist on the node.

   For example, for new Managed Servers on SOAHOST1, which is already running WLS_SOA1, select SOAHOST2. For new Managed Servers on SOAHOST2, which is already running WLS_SOA2, select SOAHOST1.

   Note: The appropriate resources must be available to run the Managed Servers concurrently during migration.

f. Verify that the **Automatic Server Migration Enabled** option is selected.

   This option enables Node Manager to start a failed server on the target node automatically.

g. Click **Save**.

h. Restart the Administration Server, Managed Servers, and Node Manager.

9. Test server migration for the new server. To test migration, perform the following steps from the node where you added the new server:

   - Abruptly stop the WLS_SOA*n* Managed Server. To do this, run `kill -9 pid` on the PID of the Managed Server. You can identify the PID of the node using the following command:

     ```
     ps -ef | grep WLS_SOAn
     ```

   - Watch the Node Manager Console for a message indicating that WLS_SOA1's floating IP has been disabled.
   - Wait for Node Manager to attempt a second restart of WLS_SOA*n*. Node Manager waits for a fence period of 30 seconds before trying this restart.
   - Once Node Manager restarts the server, stop it again. Node Manager should log a message indicating that the server will not be restarted again locally.

     Note: After a server is migrated, to fail it back to its original node or machine, stop the Managed Server from the WebLogic Administration Console and then start it again. The appropriate Node Manager will start the Managed Server on the machine to which it was originally assigned.

16.7 Scaling Out the Oracle WebCenter Content Topology

When scaling out the topology, you add new Managed Servers configured to new nodes.

Prerequisites

Before performing the steps in this section, check that you meet these requirements:

- There must be existing nodes running Managed Servers configured with Oracle Fusion Middleware within the topology.
- The new node can access the existing home directories for WebLogic Server and Fusion Middleware. (Use the existing installations in shared storage for creating a new Managed Server. You do not need to install WebLogic Server or Fusion Middleware binaries in a new location, but you do need to run `pack` and `unpack` to bootstrap the domain configuration in the new node.)
- When an *`ORACLE_HOME`* or *`WL_HOME`* is shared by multiple servers in different nodes, it is recommended that you keep 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. See the following steps.
- The new server can use a new individual domain directory or, if the other Managed Servers domain directories reside on shared storage, reuse the domain directories on those servers.

> **Note:** A shared domain directory for a Managed Server with Content Server does not work because certain files within the domain, such as intradoc.cfg, are specific to each node. To prevent issues with node-specific files, use a local (per node) domain directory for each Oracle WebCenter Content and Oracle WebCenter Content: Inbound Refinery Managed Server.

The scale-out procedure depends on the topology component:

- Section 16.7.1, "Scale-Out Procedure for Oracle WebCenter Content"
- Section 16.7.2, "Scale-Out Procedure for Oracle WebCenter Content: Imaging"
- Section 16.7.3, "Scale-Out Procedure for Oracle SOA Suite"

16.7.1 Scale-Out Procedure for Oracle WebCenter Content

To scale out the Oracle WebCenter Content servers in the enterprise deployment topology:

> **Note:** These steps are based on the assumption that you are adding a new WebCenter Content server to node *n*, where no Managed Server was running previously.

1. On the new node, mount the existing Middleware home, which should include the Oracle WebCenter Content installation and 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 on WCCHOST*n*:

   ```
   cd ORACLE_COMMON_HOME/oui/bin/

   ./attachHome.sh -jreLoc ORACLE_BASE/product/fmw/jrockit_160_version
   ```

   To update the Middleware home list, create (or edit, if another WebLogic Server installation exists in the node) the *MW_HOME*`/bea/beahomelist` file, and add *ORACLE_BASE*`/product/fmw` to it.
3. Log in to the WebLogic Server 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 address to map the IP of the node that is being used for scale-out.
6. Use the WebLogic Server Administration Console to clone WLS_WCC1 into a new Managed Server. Name it WLS_WCC*n*, where *n* is a number.

   > **Note:** These steps assume that you are adding a new server to node *n*, where no Managed Server was running previously.

7. Assign the host name or IP of WCCHOST*n* to use for the new Managed Server as the listen address of the Managed Server.
8. Assign a WLS_WCC*n* Managed Server to the newly created machine.

   Select a WLS_WCC*n* server from the list of available servers, and click **Finish**.
9. Run the `pack` command on SOAHOST1 to create a template pack:

   > **Note:** You need to do this step and the next two steps because the domain directory for Managed Servers has not yet been created on WCCHOST*n*.

   ```
   cd ORACLE_COMMON_HOME/common/bin

   ./pack.sh -managed=true -domain=ORACLE_BASE/admin/domain_name/aserver/domain_
   name -template=edgdomaintemplateScaleWCC.jar -template_name=edgdomain_
   templateScaleWCC
   ```

10. Run the following command on SOAHOST1 to copy the created template file to WCCHOST*n*:

    ```
    scp edgdomaintemplateScaleWCC.jar oracle@WCCHOSTn:/ORACLE_COMMON_
    HOME/common/bin
    ```

11. Run the `unpack` command on WCCHOST*n* to unpack the template in the Managed Server domain directory:

    ```
    cd ORACLE_COMMON_HOME/common/bin

    ./unpack.sh -domain=ORACLE_BASE/admin/domain_name/mserver/domain_name
    -template=edgdomaintemplateScaleWCC.jar -app_dir=ORACLE_BASE/admin/domain_
    name/mserver/applications
    ```

12. Run the following commands on WCCHOST*n* to start Node Manager:

    ```
    CD WL_HOME/server/bin
    ./startNodeManager.sh
    ```

13. Disable host name verification for the new Managed Server.

    Before you can start and verify the WLS_WCC*n* Managed Server, you must disable host name verification. You can re-enable it after you have configured server certificates for the communication between the Administration Server and Node Manager in WCCHOST*n*. If the source server from which the new one has been cloned had already disabled host name verification, these steps are not required (the host name verification setting is propagated to the cloned server).

    For more information, see Section 8.4.5, "Disabling Host Name Verification."

14. Start the new Managed Server, WLS_WCC*n*, from the WebLogic Server Administration Console:

    a. Expand the **Environment** node in the **Domain Structure** tree on the left.

    b. Click **Servers**.

    c. On the Summary of Server page, open the **Control** tab.

    d. Select the WLS_WCC*n* server and then click **Start**.

    e. Verify that the server status is reported as `Running` in the Administration Console.

15. Configure the new Managed Server, WLS_WCC*n*:

    a. Log in to WLS_WCC*n* at `http://WCCHOSTn:16200/cs` using your WebLogic Server administration user name and password. The WebCenter Content Configuration page opens.

    Note: At the end of page you should see this text: `Since the Revisions table in the database is not empty, you are not able to configure this server as a new instance. You may only configure this server to be a node in an existing cluster.`

b. Change the following values on the server configuration page. Make sure that Cluster Node Identifier is set to match your Managed Server ID, such as WLS_WCC*n*.

- **Content Server Instance Folder:** Set this to *ORACLE_BASE*/admin/*domain_name*/*wcc_cluster_name*/cs.

- **Native File Repository Location:** Set this to *ORACLE_BASE*/admin/domain_name/*wcc_cluster_name*/cs/vault.

- **WebLayout Folder:** Set this to *ORACLE_BASE*/admin/*domain_name*/*wcc_cluster_name*/cs/weblayout.

- **User Profile:** Set this to *ORACLE_BASE*/admin/*domain_name*/*wcc_cluster_name*/cs/data/users/profiles.

c. Click **Submit** when finished, and restart the new Managed Server using the WebLogic Server Administration Console.

16. Add the WLS_WCC*n* listen addresses to the list of allowed hosts in Oracle WebCenter Content:

a. Open the file *ORACLE_BASE*/admin/domain_name/wcc_cluster_name/cs/config/config.cfg in a text editor.

b. Add the WLS_WCC*n* listen addresses to the list of addresses that are allowed to connect to Oracle WebCenter Content.

c. Save the modified config.cfg file and restart the Oracle WebCenter Content servers for the changes to take effect.

The servers and listen addresses to be added in the Content Server pool follow:

- WCCHOST1:4444
- WCCHOST2:4444
- WCCHOST3:4444

For more information, see Section 11.17, "Creating a Connection to Oracle WebCenter Content Server."

17. Test the WLS_WCC*n* Managed Server by accessing the application on the LBR (https://wcc.mycompany.com/cs). The application should be functional.

Note: The HTTP Servers in the topology should round-robin requests to the new added server (a few requests, depending on the number of servers in the cluster, may be required to hit the new server). Its is not required to add all servers in a cluster to the WebLogicCluster directive in the Oracle HTTP Server *_vh.conf files. However routing to new servers in the cluster will take place only if at least one of the servers listed in the WebLogicCluster directive is running.

16.7.2 Scale-Out Procedure for Oracle WebCenter Content: Imaging

To scale out the Imaging servers in the enterprise deployment topology:

1. On the new node, mount the existing Middleware home, which should include the Oracle WebCenter Content installation and (optionally, if the domain directory for Managed Servers in other nodes resides on shared storage) 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 on WCCHOST*n*:

   ```
   cd ORACLE_COMMON_HOME/oui/bin/

   ./attachHome.sh -jreLoc ORACLE_BASE/product/fmw/jrockit_160_version
   ```

   To update the Middleware home list, create (or edit, if another WebLogic Server installation exists in the node) the *MW_HOME*`/bea/beahomelist` file, and add *ORACLE_BASE*`/product/fmw` to it.
3. Log in to the WebLogic Server 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 address to map the IP of the node that is being used for scale-out.
6. Use the WebLogic Server Administration Console to clone WLS_IMG1 into a new Managed Server. Name it WLS_IMG*n*, where *n* is a number.

   Note: These steps assume that you are adding a new server to node *n*, where no Managed Server was running previously.

7. Assign the host name or IP to use for the new Managed Server for the listen address of the Managed Server. If you are planning to use server migration for this server (which Oracle recommends), this should be the VIP (also called a floating IP) for the server. This VIP should be different from the one used for the existing Managed Server.

   Note: You must enable a VIP on node *n*, and you must also correctly resolve the host names in the network system used by the topology (either by DNS server or host resolution). To enable the VIPs, follow the example described in Section 9.2, "Enabling SOAHOST1VHN1 on SOAHOST1 and SOAHOST2VHN1 on SOAHOST2."

   Also, assign the newly created server to the machine you added in the step 4. Without this, the machine name of the cloned server will remain.

8. Create a JMS server for Oracle WebCenter Content: Imaging on the new Managed Server:

   a. Use the WebLogic Server Administration Console to first create a new persistent store for the new IMGJMSServer (which will be created in a later step) and name it, for example, `IMGJMSFileStore_N`. Specify the path for the store as recommended in Section 4.3, "About Recommended Locations for the Different Directories," as the directory for the JMS persistent stores:

      ORACLE_BASE/admin/*domain_name*/*cluster_name*/jms

      Note: This directory must exist before the Managed Server is started or the start operation will fail.

   b. Create a new JMS server for; for example, IMGJMSServer_N. Use the IMGJMSFileStore_N created above for this JMS server. Target the IMGJMSServer_N server to the recently created Managed Server (WLS_IMG*n*).

9. Run the `pack` command on SOAHOST1 to create a template pack:

   Note: If the domain directory for other Managed Servers resides on a shared directory, this step and the next two steps are not required. Instead, the new nodes mount the already existing domain directory and use it for the newly added Managed Server.

```
cd ORACLE_COMMON_HOME/common/bin

./pack.sh -managed=true -domain=ORACLE_BASE/admin/domain_name/aserver/domain_
name -template=edgdomaintemplateScaleIMG.jar -template_name=edgdomain_
templateScaleIMG
```

10. Run the following command on SOAHOST1 to copy the created template file to WCCHOST*n*:

    Note: If the new host, WCCHOST*n* or SOAHOST*n*, will use the same MW_HOME as SOAHOST1, then this step is not required.

```
scp edgdomaintemplateScaleIMG.jar oracle@WCCHOSTn:/ORACLE_COMMON_HOME/
common/bin
```

11. Run the `unpack` command on WCCHOST*n* to unpack the template in the Managed Server domain directory:

```
cd ORACLE_COMMON_HOME/common/bin

./unpack.sh -domain=ORACLE_BASE/admin/domain_name/mserver/domain_name
-template= edgdomaintemplateScaleIMG.jar -app_dir=ORACLE_BASE/admin/domain_
name/mserver/applications
```

12. Disable host name verification for the new Managed Server.

    Before you can start and verify the WLS_WCC*n* Managed Server, you must disable host name verification. You can re-enable it after you have configured server certificates for the communication between the Administration Server and Node Manager in WCCHOST*n*. If the source server from which the new one has been cloned had already disabled host name verification, these steps are not required (the host name verification setting is propagated to the cloned server).

    For more information, see Section 8.4.5, "Disabling Host Name Verification."

13. Disable the `Automatic Server Migration Enabled` option:

    a. In the Domain Structure tree on the left of the WebLogic Server Administration Console, expand **Environment**, and select **Servers**

    b. Select the WLS_IMG*n* Managed Server, and click the **Migration** tab.

    c. Unselect **Automatic Server Migration Enabled**.

    d. Click **Save & Activate Changes**.

14. Start Node Manager on the new node if not already started. Run the following commands on WCCHOST*n* to start Node Manager:

    ```
    CD WL_HOME/server/bin
    ./startNodeManager.sh
    ```

15. Start the new Managed Server, WLS_IMG*n*, from Fusion Middleware Control and make sure it is running.

16. Add the new Imaging server listen addresses to the list of allowed hosts in Oracle WebCenter Content Server. Follow the steps in Section 11.16, "Adding the Imaging Server Listen Addresses to the List of Allowed Hosts in Oracle WebCenter Content," to add the new server to the `SocketHostNameSecurityFilter` configuration for Oracle WebCenter Content.

17. Restart all Oracle WebCenter Content Managed Servers.

    Note: Make sure that the host names of all Imaging Managed Servers have been added to the `SocketHostNameSecurityFilter` parameter list in the *`ORACLE_BASE`*`/admin/`*`domain_name`*`/wcc_`*`cluster_name`*`/cs/config/config.cfg` file.

18. Test the WLS_IMG*n* Managed Server by accessing the application on the LBR (`https://wcc.mycompany.com/imaging`). The application should be functional.

    Note: The HTTP Servers in the topology should round-robin requests to the new added server (a few requests, depending on the number of servers in the cluster, may be required to hit the new server). Its is not required to add all servers in a cluster to the WebLogicCluster directive in the Oracle HTTP Server `*_vh.conf` files. However routing to new servers in the cluster will take place only if at least one of the servers listed in the WebLogicCluster directive is running.

19. Configure server migration for the newly added server.

    It is assumed that the source server from which the new one has been cloned (here WLS_IMG*n*) already had server migration configured. If this is the case, the following steps are not required because the server migration settings are propagated to the cloned server.

    If host-name verification certificates have been set up for Node Manager, then, as a prerequisite, you should perform these steps for the newly created server. For more information, see Section 13.3, "Enabling Host Name Verification Certificates for Node Manager."

    Note: Since this new node uses an existing shared storage installation, the node already is using a Node Manager and an environment configured for server migration that includes netmask, interface, wlsifconfig script superuser privileges, and so on. Verify the privileges defined in the new node to make sure server migration will work. For more information about privilege requirements, see Chapter 14, "Configuring Server Migration for an Enterprise Deployment."

    To configure server migration:

    a. Log in to the WebLogic Server Administration Console.

    b. In the Domain Structure window, expand the **Environment** node and then click **Servers**.

    c. On the Summary of Servers page, click the name of the server (represented as a hyperlink) in the **Name** column of the table for which you want to configure migration.

    d. On the settings page for the selected server, open the **Migration** subtab.

    e. In the **Available** field of the Migration Configuration section, select the machines to which to allow migration and click the right arrow.

    Note: Specify the least-loaded machine as the migration target for the new server. The required capacity planning must be completed so that this node has enough available resources to sustain an additional Managed Server.

    f. Choose the **Automatic Server Migration Enabled** option. This enables Node Manager to start a failed server on the target node automatically.

    g. Click **Save**.

    h. Restart the Administration Server, Managed Servers, and Node Manager.

20. Test server migration for the new server. To test migration, perform the following steps from the node where you added the new server:

    - Abruptly stop the WLS_IMG*n* Managed Server. To do this, run `kill -9 pid` on the PID of the Managed Server. You can identify the PID of the node using the following command:

      ```
      ps -ef | grep WLS_IMGn
      ```

- Watch the Node Manager Console for a message indicating that WLS_IMG*n*'s floating IP has been disabled.
- Wait for Node Manager to attempt a second restart of WLS_IMG*n*. Node Manager waits for a fence period of 30 seconds before trying this restart.
- Once Node Manager restarts the server, stop it again. Node Manager should log a message indicating that the server will not be restarted again locally.

Note: After a server is migrated, to fail it back to its original node or machine, stop the Managed Server from the WebLogic Server Administration Console and then start it again. The appropriate Node Manager will start the Managed Server on the machine to which it was originally assigned.

16.7.3 Scale-Out Procedure for Oracle SOA Suite

To scale out the Oracle SOA Suite servers in the topology:

Note: To scale out the Oracle SOA Suite subsystem used by Oracle WebCenter Content: Imaging, refer to the Oracle SOA Suite enterprise deployment topology documentation.

1. On the new node, mount the existing Middleware home, which should include the Oracle SOA Suite installation and 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, run the following commands on SOAHOST*n*:

   ```
   cd ORACLE_COMMON_HOME/oui/bin/

   ./attachHome.sh -jreLoc ORACLE_BASE/product/fmw/jrockit_160_version
   ```

   To update the Middleware home list, create (or edit, if another WebLogic Server installation exists in the node) the *MW_HOME*`/bea/beahomelist` file, and add *ORACLE_BASE*`/product/fmw` to it.
3. Log in to the WebLogic Server 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 address to map the IP of the node that is being used for scale-out.
6. Use the WebLogic Server Administration Console to clone WLS_SOA1 into a new Managed Server. Name it WLS_SOA*n*, where *n* is a number.

Note: These steps assume that you are adding a new server to node *n*, where no Managed Server was running previously.

7. Assign the host name or IP to use for the new Managed Server for the listen address of the Managed Server.

   If you are planning to use server migration for this server (which Oracle recommends), this should be the VIP (also called a floating IP) for the server. This VIP should be different from the one used for the existing Managed Server.

8. Run the `pack` command on SOAHOST1 to create a template pack:

   Note: If the domain directory for other Managed Servers resides on a shared directory, this step is not required. Instead, the new nodes mount the already existing domain directory and use it for the newly added Managed Server.

   ```
   cd ORACLE_COMMON_HOME/common/bin

   ./pack.sh -managed=true -domain=ORACLE_BASE/admin/domain_name/aserver/domain_
   name -template=edgdomaintemplateScaleSOA.jar -template_name=edgdomain_
   templateScaleSOA
   ```

9. Run the following command on SOAHOST1 to copy the created template file to SOAHOST*n*:

   ```
   scp edgdomaintemplateScaleSOA.jar oracle@SOAHOSTn:/ORACLE_COMMON_
   HOME/common/bin
   ```

10. Run the `unpack` command on SOAHOST*n* to unpack the template in the Managed Server domain directory:

    ```
    cd ORACLE_COMMON_HOME/common/bin

    ./unpack.sh -domain=ORACLE_BASE/admin/domain_name/mserver/domain_name
    -template= edgdomaintemplateScaleSOA.jar -app_dir=ORACLE_BASE/admin/domain_
    name/mserver/applications
    ```

11. Create JMS servers for Oracle SOA Suite and UMS on the new Managed Server:

    a. Use the WebLogic Server Administration Console to create a new persistent store for the new SOAJMSServer and name it, for example, `SOAJMSFileStore_N`. Specify the path for the store. This should be a directory on shared storage as recommended in Section 4.3, "About Recommended Locations for the Different Directories":

       ```
       ORACLE_BASE/admin/domain_name/cluster_name/jms
       ```

       Note: This directory must exist before the Managed Server is started or the start operation fails.

    b. Create a new JMS server for Oracle SOA Suite (for example, SOAJMSServer_N). Use SOAJMSFileStore_N for this JMS server. Target the SOAJMSServer_N server to the recently created Managed Server (WLS_SOA*n*).

    c. Create a new persistence store for the new UMSJMSServer (for example, `UMSJMSFileStore_N`). Specify the path for the store. This should be a directory on shared storage, as recommended in Section 4.3, "About Recommended Locations for the Different Directories":

       ```
       ORACLE_BASE/admin/domain_name/cluster_name/jms
       ```

Note: This directory must exist before the Managed Server is started or the start operation fails. You can also assign SOAJMSFileStore_N as the store for the new UMS JMS servers. For the purpose of clarity and isolation, individual persistent stores are used in the following steps.

d. Create a new JMS server for UMS (for example, UMSJMSServer_N). Use the UMSJMSFileStore_N for this JMS server. Target the UMSJMSServer_N server to the recently created Managed Server (WLS_SOA*n*).

e. Update the subdeployment targets for the SOA JMS Module to include the recently created SOA JMS server. To do this, expand the **Services** node in the **Domain Structure** tree on the left of the WebLogic Server Administration Console, and then expand the **Messaging** node. Click **JMS Modules**. The JMS Modules page appears. Click **SOAJMSModule** (represented as a hyperlink in the Names column of the table). The Settings page for SOAJMSModule appears. Open the **SubDeployments** tab. The subdeployment module for SOAJMS appears.

Note: This subdeployment module name is a random name in the form of `SOAJMSServer`*`XXXXXX`* resulting from the Configuration Wizard JMS configuration for the first two servers (WLS_SOA1 and WLS_SOA2).

Click the **SOAJMSServer*XXXXXX*** subdeployment. Add the new JMS server for Oracle SOA Suite called SOAJMSServer_N to this subdeployment. Click **Save**.

f. Update the subdeployment targets for the UMSJMSSystemResource to include the recently created UMS JMS server. To do this, expand the **Services** node in the **Domain Structure** tree on the left of the WebLogic Server Administration Console, and then expand the **Messaging** node. Click **JMS Modules**. The JMS Modules page appears. Click **UMSJMSSystemResource** (represented as a hyperlink in the Names column of the table). The Settings page for UMSJMSSystemResource appears. Open the **SubDeployments** tab. The subdeployment module for UMSJMS appears.

Note: This subdeployment module name is a random name in the form of `UCMJMSServer`*`XXXXXX`* resulting from the Configuration Wizard JMS configuration for the first two servers (WLS_SOA1 and WLS_SOA2).

Click the **UMSJMSServer*XXXXXX*** subdeployment. Add the new JMS server for UMS called UMSJMSServer_N to this subdeployment. Click **Save**.

12. Configure a TX persistent store for the new server. This should be a location visible from other nodes as indicated in the recommendations about shared storage (see Section 4.3, "About Recommended Locations for the Different Directories").

From the Administration Console, select the server name (WLS_SOA*n*) in the **Services** tab. Under **Default Store**, in **Directory**, enter the path to the folder where you want the default persistent store to store its data files:

`ORACLE_BASE/admin/`*`domain_name`*`/`*`soa_cluster_name`*`/tlogs`

13. Disable host name verification for the new Managed Server. Before you can start and verify the WLS_SOA*n* Managed Server, you must disable host name verification. You can re-enable it after you have configured server certificates for the communication between the Administration Server and Node Manager in SOAHOST*n*. If the source server from which the new one has been cloned had already disabled host name verification, these steps are not required (the host name verification setting is propagated to the cloned server).

    To disable host name verification:

    a. In Oracle Enterprise Manager Fusion Middleware Control, select **Oracle WebLogic Server Administration Console**.

    b. Expand the **Environment** node in the Domain Structure window.

    c. Click **Servers**.

    d. On the Summary of Servers page, select WLS_SOA*n* in the Names column of the table.

    e. On the settings page for the server, open the **SSL** tab.

    f. Expand the **Advanced** section of the page.

    g. Click **Lock & Edit**.

    h. Set host name verification to `None`.

    i. Click **Save**.

14. Start Node Manager on the new node. To start Node Manager, use the installation in shared storage from the already existing nodes and then start Node Manager by passing the host name of the new node as a parameter as follows (on SOAHOST*n*):

    ```
    WL_HOME/server/bin/startNodeManager.sh New_Node_IP
    ```

    Note: If you used the paths shown in Chapter 6.3.1, "Installing Oracle WebLogic Server and Creating the Middleware Home," *WL_HOME* would be *ORACLE_BASE*`/product/fmw/wlserver_10.3`.

15. Start the new Managed Server, WLS_SOA*n*, from the WebLogic Server Administration Console, and make sure it is running.

16. Test the WLS_SOA*n* Managed Server by accessing the application on the LBR (`http://soainternal.mycompany.com/soa-infra`). The application should be functional.

    Note: The HTTP Servers in the topology should round-robin requests to the new added server (a few requests, depending on the number of servers in the cluster, may be required to hit the new server). Its is not required to add all servers in a cluster to the WebLogicCluster directive in the Oracle HTTP Server `*_vh.conf` files. However routing to new servers in the cluster will take place only if at least one of the servers listed in the WebLogicCluster directive is running.

17. Configure server migration for the newly added server.

    Note: Since this new node uses an existing shared storage installation, the node already is using a Node Manager and an environment configured for server migration that includes netmask, interface, wlsifconfig script superuser privileges, and so on. The floating IP for the new Oracle SOA Suite Managed Server is already present in the new node.

    To configure server migration:

    a. Log in to the WebLogic Server Administration Console.

    b. In the Domain Structure window, expand the **Environment** node and then click **Servers**.

    c. On the Summary of Servers page, click the name of the server (represented as a hyperlink) in Name column of the table for which you want to configure migration.

    d. On the settings page for the selected server, open the Migration subtab.

    e. In the **Available** field of the Migration Configuration section, select the machines to which to allow migration and click the right arrow.

    Note: Specify the least-loaded machine as the migration target for the new server. The required capacity planning must be completed so that this node has enough available resources to sustain an additional Managed Server.

    f. Choose the **Automatic Server Migration Enabled** option. This enables Node Manager to start a failed server on the target node automatically.

    g. Click **Save**.

    h. Restart the Administration Server, Managed Servers, and Node Manager.

18. Test server migration for the new server. To test migration, perform the following steps from the node where you added the new server:

    - Abruptly stop the WLS_SOA*n* Managed Server. To do this, run `kill -9 pid` on the PID of the Managed Server. You can identify the PID of the node using the following command:

      ```
      ps -ef | grep WLS_SOAn
      ```

    - Watch the Node Manager Console for a message indicating that WLS_SOA*n*'s floating IP has been disabled.
    - Wait for Node Manager to attempt a second restart of WLS_SOA*n*. Node Manager waits for a fence period of 30 seconds before trying this restart.
    - Once Node Manager restarts the server, stop it again. Node Manager should log a message indicating that the server will not be restarted again locally.

Note: After a server is migrated, to fail it back to its original node or machine, stop the Managed Server from the WebLogic Server Administration Console and then start it again. The appropriate Node Manager will start the Managed Server on the machine to which it was originally assigned.

16.8 Verifying Manual Failover of the Administration Server

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:

- Section 16.8.1, "Assumptions and Procedure"
- Section 16.8.2, "Validating Access to SOAHOST2 Through the Load Balancer"
- Section 16.8.3, "Failing the Administration Server Back to SOAHOST1"

16.8.1 Assumptions and Procedure

Please note the following assumptions:

- The Administration Server is configured to listen on ADMINVHN, and not on ANY address. See step 13 in Section 8.3, "Running the Configuration Wizard on SOAHOST1 to Create a Domain."
- The Administration Server is failed over from SOAHOST1 to SOAHOST2, and the two nodes have these IP addresses:
  - SOAHOST1: 100.200.140.165
  - SOAHOST2: 100.200.140.205
  - ADMINVHN: 100.200.140.206. This is the VIP where the Administration Server is running, assigned to eth*X*:*Y*.
- The domain directory where the Administration Server is running in SOAHOST1 is on a shared storage and is mounted also from SOAHOST2.
- Oracle WebLogic Server and Oracle Fusion Middleware components have been installed in SOAHOST2 as described in Chapter 6, "Installing the Software for an Enterprise Deployment" (that is, the same paths for *ORACLE_HOME* and *MW_HOME* that exist on SOAHOST1 are also available on SOAHOST2).

Procedure

The following procedure shows how to fail over the Administration Server to a different node (SOAHOST2):

1. Stop the Administration Server if it is running.
2. Migrate the IP address to the second node:
   a. Run the following command as `root` on SOAHOST1 (where *X*:*Y* is the current interface used by ADMINVHN):

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

   b. Run the following command on SOAHOST2:

      ```
      /sbin/ifconfig interface:index IP_address netmask netmask
      ```

For example:

```
/sbin/ifconfig eth0:1 100.200.140.206 netmask 255.255.255.0
```

Note: Make sure that the netmask and interface to be used match the available network configuration in SOAHOST2. Also, make sure that the location of the Administration Server application directory is mounted as described in Section 4.3, "About Recommended Locations for the Different Directories."

c. Update the routing tables on SOAHOST2 using `arping`; for example:

```
/sbin/arping -q -U -c 3 -I eth0 100.200.140.206
```

3. Start Node Manager in SOAHOST2, as described in Section 8.4.2, "Starting Node Manager on SOAHOST1."
4. Start the Administration Server on SOAHOST2, as described in Section 8.4.3, "Starting the Administration Server on SOAHOST1."
5. Test that you can access the Administration Server on SOAHOST2 as follows:
   a. Ensure that you can access the WebLogic Server Administration Console at `http://ADMINVHN:7001/console`.
   b. Check that you can access and verify the status of components in Oracle Enterprise Manager Fusion Middleware Control at `http://ADMINVHN:7001/em`.

16.8.2 Validating Access to SOAHOST2 Through the Load Balancer

Perform the same steps as in Section 8.5.5, "Validating Access Through the Load Balancer." This is to check that you can access the Administration Server when it is running on SOAHOST2.

16.8.3 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 again. To do this, migrate ADMINVHN back to the SOAHOST1 node as follows:

1. Make sure the Administration Server is not running.
2. Run the following command on SOAHOST2.

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

3. Run the following command on SOAHOST1:

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

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

4. Update the routing tables using `arping`. Run the following command from SOAHOST1:

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

5. Start the Administration Server again on SOAHOST1, as described in Section 8.4.3, "Starting the Administration Server on SOAHOST1."
6. Test that you can access the WebLogic Server Administration Console at `http://ADMINVHN:7001/console`.
7. Check that you can access and verify the status of components in Oracle Enterprise Manager Fusion Middleware Control at `http://ADMINVHN:7001/em`.

16.9 Performing Backups and Recoveries in Oracle WebCenter Content Enterprise Deployments

Table 16–2 lists the static artifacts to back up in the Oracle WebCenter Content 11*g* enterprise deployment.

Table 16–2 Static Artifacts to Back Up in the Oracle WebCenter Content 11g Enterprise Deployment

Type	Host	Location	Tier
ORACLE HOME (DB)	CUSTDBHOST1 and CUSTDBHOST2	The location is user defined.	Data tier
MW HOME (OHS)	WEBHOST1 and WEBHOST2	*ORACLE_BASE*/product/fmw/	Web tier
MW HOME (this includes the SOA Oracle home as well)	SOAHOST1 and SOAHOST2*	*MW_HOME* The SOA Oracle home is also under *MW_HOME*: *ORACLE_HOME*	Application tier
Installation-related files		OraInventory, *User_Home*/bea/beahomelist, oraInst.loc, oratab	N/A

* WCCHOST1 and WCCHOST2 use the binaries installed from SOAHOST1 and SOAHOST2. Backup is centralized in SOAHOST1 and SOAHOST2.

Table 16–3 lists the runtime artifacts to back up in the Oracle WebCenter Content 11*g* enterprise deployment.

Table 16–3 Runtime Artifacts to Back Up in the Oracle WebCenter Content 11g Enterprise Deployment

Type	Host	Location	Tier
Application artifacts (EAR and WAR files)	SOAHOST1, SOAHOST2, WCCHOST1, and WCCHOST2	Find the application artifacts by viewing all of the deployments through the administration console.	Application tier
Oracle SOA Suite runtime artifacts	SOAHOST1 or SOAHOST2	*ORACLE_BASE*/admin/*domain_name*/*soa_cluster_name*	Application tier
Oracle WebCenter Content runtime artifacts	WCCHOST1 or WCCHOST2	*ORACLE_BASE*/admin/*domain_name*/*wcc_cluster_name*	Application tier
Oracle WebCenter Content: Imaging runtime artifacts	WCCHOST1 or WCCHOST2	*ORACLE_BASE*/admin/*domain_name*/*img_cluster_name*	Application tier

Table 16–3 (Cont.) Runtime Artifacts to Back Up in the Oracle WebCenter Content 11g Enterprise

Type	Host	Location	Tier
Customized Managed Server configuration for Oracle WebCenter Content	WCCHOST1 or WCCHOST2	*ORACLE_BASE*/admin/*domain_name*/mserver/*domain_name*/ucm/cs/bin/intradoc.cfg and *ORACLE_BASE*/admin/*domain_name*/mserver/*domain_name*/bin/server_migration/wlsifconfig.sh	Application tier
Customized Managed Server configuration for Oracle SOA Suite	SOAHOST1 or SOAHOST2	If using UMS: *DOMAIN_HOME*/servers/*server_name*/tmp/_WL_user/*ums_driver_name*/*/configuration/driverconfig.xml (where * represents a directory whose name is randomly generated by WebLogic Server during deployment; for example, 3682yq) and *ORACLE_BASE*/admin/*domain_name*/mserver/*domain_name*/bin/server_migration/wlsifconfig.sh	Application tier
Oracle HTTP Server instance home	WEBHOST1 and WEBHOST2	*ORACLE_BASE*/admin/*instance_name*	Web tier
Oracle RAC databases	CUSTDBHOST1 and CUSTDBHOST2	The location is user-defined.	Data tier

For more information on backup and recovery of Oracle Fusion Middleware components, see *Oracle Fusion Middleware Administrator's Guide*.

16.10 Preventing Timeouts for SQLNet Connections

Much of the production enterprise 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 Oracle RAC, set this parameter in all of the Oracle home directories.

16.11 Configuring Oracle Web Service Manager Security Policies for Oracle WebCenter Content and Imaging Services

When first installed, the Oracle WebCenter Content and Imaging web services are configured with no Oracle Web Service Manager (OWSM) security policies applied. When no security policies are applied, the services leverage the basic HTTP authentication mechanism, where user credentials (user ID and password) are transmitted in the web service HTTP message header. Oracle recommends using the appropriate Oracle WSM policy enforcements instead of basic HTTP authentication. To configure Oracle WSM security policies for Oracle WebCenter Content and Imaging web services, follow the steps in the *Oracle WebCenter Content Developer's Guide for Imaging* and the *Oracle WebCenter Content Installation Guide*.

16.12 Troubleshooting the Oracle WebCenter Content Enterprise Deployment Topology

This section covers the following topics:

- Section 16.12.19, "Caching and Locking Files"
- Section 16.12.20, "Modifying Upload and Stage Directories for Applications Deployed Remotely"

16.12.1 Page Not Found When Accessing soa-infra Application Through Load Balancer

Problem: A 404 `page not found` message is displayed in the web browser when you try to access the soa-infra application using the load balancer address. The error is intermittent and Oracle SOA Suite servers appear as `Running` in the WebLogic Server Administration Console.

Solution: Even when the Oracle SOA Suite Managed Servers are up and running, some of the applications contained in them can be in `Admin`, `Prepared` or other states different from `Active`. The soa-infra application may be unavailable while the Oracle SOA Suite server is running. Check the Deployments page in the Administration Console to verify the status of the soa-infra application. It should be in the `Active` state. Check the Oracle SOA Suite server's output log for errors pertaining to the soa-infra application and try to start it from the Deployments page in the Administration Console.

16.12.2 Soa-infra Application Fails to Start Due to Deployment Framework Issues (Coherence)

Problem: The soa-infra application fails to start after changes to the Coherence configuration for deployment have been applied. The Oracle SOA Suite server output log reports the following:

```
Cluster communication initialization failed. If you are using multicast, Please
make sure multicast is enabled on your network and that there is no interference
on the address in use. Please see the documentation for more details.
```

Solutions:

1. When using multicast instead of unicast for cluster deployments of SOA composites, you may get a message similar to the preceding one if a multicast conflict arises when starting the soa-infra application (that is, starting the Managed Server on which Oracle SOA Suite runs). These messages, which occur when Oracle Coherence throws a run-time exception, also include the details of the exception itself. If such a message appears, check the multicast configuration in your network. Verify that you can ping multicast addresses. In addition, check for other clusters that may have the same multicast address but have a different cluster name in your network, as this may cause a conflict that prevents soa-infra from starting. If multicast is not enabled in your network, you can change the deployment framework to use unicast as described in the *Oracle Coherence Developer's Guide*.
2. When entering well-known address list for unicast (in server start parameters), make sure that the node's addresses entered for the localhost and clustered servers are correct. Error messages like the following are reported in the server's output log if any of the addresses is not resolved correctly:

   ```
   oracle.integration.platform.blocks.deploy.CompositeDeploymentCoordinatorMessage
   s errorUnableToStartCoherence
   ```

16.12.3 Incomplete Policy Migration After Failed Restart of SOA Server

Problem: The Oracle SOA Suite server fails to start through the administration console *before* setting Node Manager property `startScriptEnabled=true`. The server does not come up after the property is set either. The Oracle SOA Suite server output log reports the following:

```
SEVERE: <.> Unable to Encrypt data
Unable to Encrypt data.
Check installation/post-installation steps for errors. Check for errors during SOA
server startup.

ORABPEL-35010
 .
Unable to Encrypt data.
Unable to Encrypt data.
Check installation/post-installation steps for errors. Check for errors
 during SOA server startup.
 .
 at
oracle.bpel.services.common.util.EncryptionService.encrypt(EncryptionService.java:
56)
...
```

Solution: Incomplete policy migration results from an unsuccessful start of the first Oracle SOA Suite server in a cluster. To enable full migration, edit the `<jazn-policy>` element the system-jazn-data.xml file to grant permission to bpm-services.jar:

```
<grant>
  <grantee>
    <codesource>
<url>file:${oracle.home}/soa/modules/oracle.soa.workflow_11.1.1/bpm-services.jar
</url>
    </codesource>
  </grantee>
  <permissions>
    <permission>
      <class>java.security.AllPermission</class>
    </permission>
  </permissions>
</grant>
```

16.12.4 SOA, Oracle WebCenter Content, or Imaging Servers Fail to Start Due to Maximum Number of Processes Available in Database

Problem: An Oracle SOA Suite, Oracle WebCenter Content, or Imaging server fails to start. The domain has been extended for new types of Managed Server (for example, Oracle WebCenter Content extended for Imaging) or the system has been scaled up (added new servers of the same type). The Oracle SOA Suite, Oracle WebCenter Content, or Imaging server output log reports the following:

```
<Warning> <JDBC> <BEA-001129> <Received exception while creating connection for
pool "SOADataSource-rac0": Listener refused the connection with the following
error:

ORA-12516, TNS:listener could not find available handler with matching protocol
stack >
```

Solution: Verify the number of processes in the database and adjust accordingly. As the SYS user, issue the SHOW PARAMETER command:

```
SHOW PARAMETER processes
```

Set the initialization parameter using the following SQL command:

```
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. For information about parameter files, server parameter files, and how to change parameter value, see the *Oracle Database Administrator's Guide*.

16.12.5 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/edg_domain/aserver/edg_
domain/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.lok.

16.12.6 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 **Activate Changes** is clicked:

```
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, provide user name and password information in the server start configuration in the Administration Console for the specific server whose configuration was being changed.

16.12.7 SOA or Imaging Server Not Failed Over After Server Migration

Problem: After reaching the maximum restart attempts by local Node Manager, Node Manager in the failover node tries to restart it, but the server does not come up. The server seems to be failed over as reported by Node Manager's output information. The VIP used by the Oracle SOA Suite or Oracle WebCenter Content: Imaging server is not enabled in the failover node after Node Manager tries to migrate it (if config in the failover node does not report the VIP in any interface). Executing the command `sudo ifconfig $INTERFACE $ADDRESS $NETMASK` does not enable the IP in the failover node.

Solution: The rights and configuration for `sudo` execution should not prompt for a password. Verify the configuration of `sudo` with your system administrator so that `sudo` works without a password prompt.

16.12.8 SOA or Imaging Server Not Reachable from Browser After Server Migration

Problem: Server migration is working (Oracle SOA Suite or Oracle WebCenter Content: Imaging server is restarted in the failed over node), but the *`Virtual_Hostname`*`:8001/soa-infra` URL cannot be accessed in the web browser. The server has been *killed* in its original host, and Node Manager in the failover node reports that the VIP has been migrated and the server started. The VIP used by the Oracle SOA Suite or Imaging server cannot be pinged from the client's node (that is, the node where the browser is being used).

Solution: The `arping` command executed by Node Mnager to update ARP caches did not broadcast the update properly. In this case, the node is not reachable to external nodes. Either update the nodemanager.properties file to include the MACBroadcast or execute a manual arping:

```
/sbin/arping -b -q -c 3 -A -I INTERFACE ADDRESS > $NullDevice 2>&1
```

Where *INTERFACE* is the network interface where the virtual IP is enabled and *ADDRESS* is the virtual IP address.

16.12.9 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 Oracle Access Manager. Log in to the Access Administration site for Oracle Access Manager, click **My Policy Domains**, click the created policy domain (**SOA_EDG**), then click the **Resources** tab, and remove the incorrect URLs.

16.12.10 Redirection of Users to Login Screen After Activating Changes in the Administration Console

Problem: After configuring OHS and LBR to access the WebLogic Server Administration Console, some activation changes cause the redirection to the login screen for the Administration 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 `wcc.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.

16.12.11 Redirection of Users to Administration Console's Home Page After Activating Changes to Oracle Access Manager

Problem: After configuring Oracle Access Manager, 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 Oracle Access Manager SSO is configured and the Administration Console is set to follow configuration changes (redirections are performed by the Administration Server when activating some changes). Activations should complete regardless of this redirection. For successive changes not to redirect, access the Administration Console, choose Preferences, then Shared Preferences, and unselect the **Follow Configuration Changes** checkbox.

16.12.12 Configured JOC Port Already in Use

Problem: Attempts to start a Managed Server that uses the Java Object Cache, such as OWSM 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.

16.12.13 Using CredentialAccessPermissions to Allow Oracle WebCenter Content: Imaging to Read Credentials from the Credential Store

Problem: Oracle WebCenter Content: Imaging creates the credential access permissions during startup and updates its local domain directory copy of the system-jazn-data.xml file. While testing the environment without an LDAP policy store being configured, the Administration Server may push manual updates to the system.jazn-data.xml file to the domain directories where the Oracle WebCenter Content: Imaging servers reside. This can cause the copy of the file to be overwritten, given rise to a variety of exceptions and errors in the restarts or access to the Oracle WebCenter Content: Imaging console.

Solution: To re-create the credential access permissions and update the Administration Server's domain directory copy of the system-jazn-data.xml file, use the `grantIPMCredAccess` command from the Oracle WebLogic Scripting Tool. To do this, start wlst.sh from the *ORACLE_HOME* associated with Oracle WebCenter Content, connect to the Administration Server, and execute the `grantIPMCredAccess()` command on WCCHOST1:

```
cd ORACLE_HOME/common/bin

./wlst.sh

wls:/offline> connect()

wls:/domain_name/serverConfig> grantIPMCredAccess()
```

Note: When connecting, provide the credentials and address for the Administration Server.

16.12.14 Improving Performance with Very Intensive Document Uploads from Oracle WebCenter Content: Imaging to Oracle WebCenter Content

Problem: If a host name-based security filter is used in Oracle WebCenter Content (config.cfg file), a high latency and performance impact may be observed in the system in the event of very intensive document uploads from Oracle WebCenter Content: Imaging to Oracle WebCenter Content. This is caused by the reverse DNS lookup which is required in Oracle WebCenter Content to allow the connections from the Imaging servers.

Solution: Using a host name-based security filter is recommended in preparation of configuring the system for disaster protection and to restore to a different host (since the configuration used is IP-agnostic when using a host name-based security filter). However, if the performance of the uploads needs to be improved, you can use an IP-based security filter instead of a host name-based filter.

To change the host name-based security filter in Oracle WebCenter Content to an IP-based filter:

1. Open the file *ORACLE_BASE*`/admin/`*domain_name*`/`*wcc_cluster_name*`/cs/config/config.cfg` in a text editor.
2. Remove or comment out the following two lines:

   ```
   SocketHostNameSecurityFilter=localhost|localhost.mycompany.com|wcchost1vhn1|
   wcchost2vhn1
   AlwaysReverseLookupForHost=Yes
   ```

3. Add the IP addresses (listen addresses) of the WLS_IMG1 and WLS_IMG2 Managed Servers (WCCHOST1VHN1 and WCCHOST2VHN1, respectively) to the `SocketHostAddressSecurityFilter` parameter list:

   ```
   SocketHostAddressSecurityFilter=127.0.0.1|0:0:0:0:0:0:0:1|X.X.X.X|Y.Y.Y.Y
   ```

   where *X.X.X.X* and *Y.Y.Y.Y* are the listen addresses of WLS_IMG1 and WLS_IMG2, respectively. (Please note that 127.0.0.1 must be included in the list as well.)
4. Save the modified `config.cfg` file and restart the Oracle WebCenter Content servers for the changes to take effect.

16.12.15 Out-of-Memory Issues on Managed Servers

Problem: You are experiencing out-of-memory issues on Managed Servers.

Solution: Increase the size of the memory heap allocated for the Java VM to at least one gigabyte:

1. Log in to the WebLogic Server Administration Console.
2. Click **Environment**, then **Servers**.
3. Click a Managed Server name.
4. Open the **Configuration** tab.
5. Open the **Server Start** tab in the second row of tabs.
6. Include the memory parameters in the Arguments box, for example:

```
-Xms256m -Xmx1024m -XX:CompileThreshold=8000 -XX:PermSize=128m
-XX:MaxPermSize=1024m
```

> **Note:** Please note that the memory parameter requirements may differ between various JVMs (Sun, JRockit, or others). See "Increasing the Java VM Heap Size for Managed Servers" in the *Oracle WebCenter Content Installation Guide* for further details.

7. Save the configuration changes.
8. Restart all running Managed Servers.

16.12.16 Regenerating the Master Password for Oracle WebCenter Content Servers

If the `cwallet.sso` file of the Oracle WebCenter Content Managed Servers domain home becomes inconsistent across the cluster, is deleted, or is accidentally overwritten by an invalid copy in the *ORACLE_BASE*`/admin/`*domain_name*`/aserver/`*domain_name*`/config/fmwconfig/` directory, you can perform these steps to regenerate the file:

1. Stop all Oracle WebCenter Content Managed Servers (WLS_WCC*x*).
2. Remove the `cwallet.sso` file from *ORACLE_BASE*`/admin/`*domain_name*`/mserver/`*domain_name*`/config/fmwconfig/`.
3. Remove the `password.hda` file from *ORACLE_BASE*`/admin/`*domain_name*`/aserver/`*wcc_cluster_name*`/cs/config/private`.
4. Start the WLS_WCC1 server in WCCHOST1.
5. Verify the creation or update of the `cwallet.sso` file in *ORACLE_BASE*`/admin/`*domain_name*`/mserver/`*domain_name*`/config/fmwconfig/` as well as the creation of the `password.hda` file in *ORACLE_BASE*`/admin/`*domain_name*`/aserver/`*wcc_cluster_name*`/cs/config/private/`.
6. Use the Oracle WebCenter Content System Properties command-line tool to update the passwords for the database.
7. Verify that the standalone Oracle WebCenter Content applications (Batchloader, System Properties, and so on) are working correctly.

8. Copy the `cwallet.sso` file from *ORACLE_BASE*`/admin/`*domain_name*`/mserver/`*domain_name*`/config/fmwconfig/` to the Administration Server's domain directory at *ORACLE_BASE*`/admin/`*domain_name*`/aserver/`*domain_name*`/config/fmwconfig/`.
9. Start the second Oracle WebCenter Content Server, and verify that the Administration Server pushes the updated `cwallet.sso` file to *ORACLE_BASE*`/admin/`*domain_name*`/mserver/`*domain_name*`/config/fmwconfig/` in WCCHOST2 and that the file is the same as created or updated by the Oracle WebCenter Content Server in WCCHOST1.
10. Verify that the standalone Oracle WebCenter Content applications (Batchloader, System Properties, and so on) are working correctly.
11. Verify that the standalone Oracle WebCenter Content applications work correctly on both nodes at the same time.

16.12.17 Logging Out from the WebLogic Server Administration Console Does Not End the User Session

When you log in to the WebLogic Server Administration Console using Oracle Access Manager single sign-on (SSO), then clicking the logout button does not end the user session. You are not redirected to the Oracle Access Manager login page, which is in accordance with the SSO logout guidelines, but rather the home page is reloaded. To truly log out, you may need to manually clean up the cookies for your web browser.

16.12.18 Transaction Timeout Error

Problem: The following transaction timeout error appears in the log:

```
Internal Exception: java.sql.SQLException: Unexpected exception while enlisting
 XAConnection java.sql.SQLException: XA error: XAResource.XAER_NOTA start()
failed on resource 'SOADataSource_soaedg_domain': XAER_NOTA : The XID
is not valid
```

Solution: Check your transaction timeout settings, and be sure that the JTA transaction timeout is less than the DataSource XA Transaction Timeout, which is less than the `distributed_lock_timeout` (at the database).

With the out-of-the-box configuration, the SOA data sources do not set XA timeout to any value. The `Set XA Transaction Timeout` configuration parameter is unchecked in the WebLogic Server Administration Console. In this case, the data sources use the domain-level JTA timeout, which is set to `30`. Also, the default `distributed_lock_timeout` value for the database is `60`. As a result, the SOA configuration works correctly for any system where transactions are expected to have lower life expectancy than such values. Adjust these values according to the transaction times your specific operations are expected to take.

16.12.19 Caching and Locking Files

Oracle WebCenter Content uses its own locking mechanism for files, so it needs to have access to those files without file-attribute caching and without locking being done by the cluster nodes. If one of the nodes accesses a certain status file and it happens to be cached, that node might attempt to run a process that another node is already working on. Or if a particular file is locked by one of the node clients, this could interfere with access by another node. Unfortunately, disabling file-attribute caching on the file share can impact performance. So it is important to disable caching and locking only on the particular folders that require it. For instance, if you are creating the share through Network File System (NFS), use `noac` and `nolock` for the mount options.

16.12.20 Modifying Upload and Stage Directories for Applications Deployed Remotely

If you are deploying applications remotely, you might need to update your `upload` and `stage` directories to absolute paths after you create the domain and unpack to the `mserver` directory. Absolute path names can prevent issues for remote deployments and confusion for deployments that use the stage mode.

The default path names for these directories follow:

```
./servers/AdminServer/upload

./servers/server_name/stage
```

A

Using Multi Data Sources with Oracle RAC

Oracle recommends using GridLink data sources when developing new Oracle RAC applications. However, if you are using legacy applications and databases that do not support GridLink data sources, refer to the information in this appendix.

This appendix provides the following topics:

- Section A.1, "About Multi Data Sources and Oracle RAC"
- Section A.2, "Typical Procedure for Configuring Multi Data Sources for an EDG Topology"

A.1 About Multi Data Sources and Oracle RAC

A multi data source provides an ordered list of data sources to use to satisfy connection requests. Normally, every connection request to this kind of multi data source is served by the first data source in the list. If a database connection test fails and the connection cannot be replaced, or if the data source is suspended, a connection is sought sequentially from the next data source on the list.

For more information about configuring Multi Data Sources with Oracle RAC, see "Using Multi Data Sources with Oracle RAC" in the *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server*.

A.2 Typical Procedure for Configuring Multi Data Sources for an EDG Topology

You configure data sources when you configure a domain. For example, when you are configuring the initial Administration domain for an Enterprise Deployment reference topology, you use the configuration wizard to define the characteristics of the domain, as well as the data sources.

The procedures for configuring the topologies in this Enterprise Deployment Guide include specific instructions for defining GridLink data sources with Oracle RAC. If you want to use Multi Data Sources instead of GridLink data sources, replace the GridLink instructions with the following:

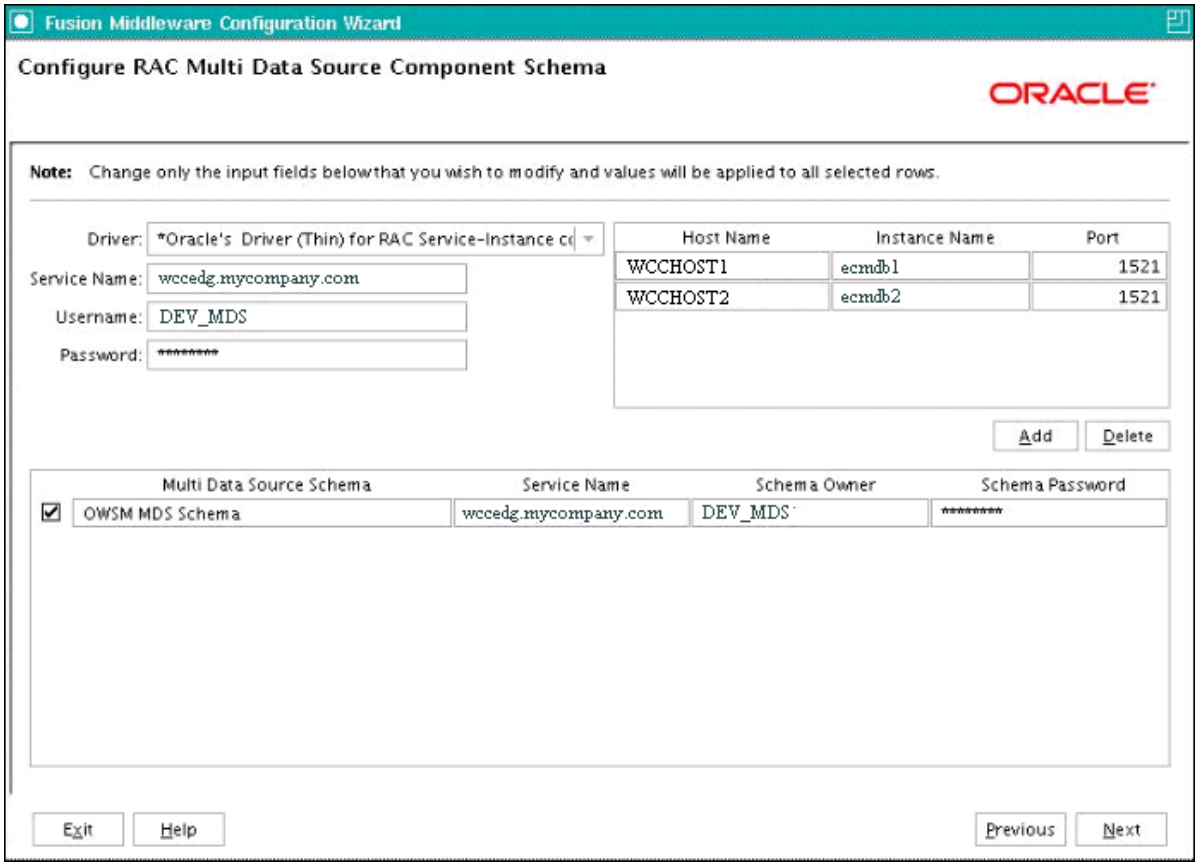
1. In the Configure JDBC Component Schema screen:
   a. Select the appropriate schemas.
   b. For the RAC configuration for component schemas, **Convert to RAC multi data source**.

c. Ensure that the following data source appears on the screen with the schema prefix when you ran the Repository Creation Utility.

d. Click **Next**.

2. The Configure RAC Multi Data Sources Component Schema screen appears (Figure A–1).

Figure A–1 Configure RAC Multi Data Source Component Schema Screen



In this screen, do the following:

a. Enter values for the following fields, specifying the connect information for the Oracle 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.

   – **Username:** Enter the complete user name for the database schema owner of the corresponding component.

     This book uses `DEV` as the prefix of user names for the database 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 Oracle RAC instance.

e. Click **Next**.

3. In the Test JDBC Data Sources screen, the connections are 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.

B

Targeting Applications and Resources to Servers

This appendix lists the applications, data sources, persistent stores, Java Messaging Service (JMS) modules, and startup and shutdown classes that are targeted to the servers in the Oracle WebLogic Server domain for Oracle WebCenter Content.

This appendix includes the following sections:

B.1 Applications and Resources for the Oracle SOA Suite Managed Servers

The following applications and resources are targeted to the Oracle SOA Suite cluster, soa_cluster, which includes these Managed Servers:

- WLS_SOA1 on SOAHOST1
- WLS_SOA2 on SOAHOST2

Applications for Oracle SOA Suite

- AqAdapter
- b2bu
- composer
- DbAdapter
- DefaultToDoTaskFlow
- DMS Application (11.1.1.1.0)
- FileAdapter
- FtpAdapter
- JmsAdapter

- MQSeriesAdapter
- OracleAppsAdapter
- OracleBamAdapter
- soa-infra
- SocketAdapter
- usermessagingdriver-email
- usermessagingserver
- worklistapp
- wsil-wls
- wsm-pm

Data Sources for Oracle SOA Suite

- `EDNDataSource`
- `EDNLocalTxDataSource`
- `mds-owsm`
- `mds-soa`
- `OraSDPMDataSource`
- `SOADataSource`
- `SOALocalTxDataSource`

Persistence Stores for Oracle SOA Suite

- `BPMJMSFileStore_auto_1`
- `BPMJMSFileStore_auto_2`
- `SOAJMSFileStore_auto_1`
- `SOAJMSServer_auto_2`
- `UMSJMSFileStore_auto_1`
- `UMSJMSFileStore_auto_2`

JMS Deployment Modules for Oracle SOA Suite

- `BPMJMSServer_auto_1`
- `BPMJMSServer_auto_2`
- `SOAJMSServer_auto_1`
- `SOAJMSServer_auto_2`
- `UMSJMSServer_auto_1`
- `UMSJMSServer_auto_2`

Startup and Shutdown Classes for Oracle SOA Suite

- `oracle.jrf.AppContextStartup`
- `oracle.dms.wls.DMSStartup`
- `oracle.dms.wls.DMSShutdown`

- oracle.as.jmx.framework.wls.spi.StartupListener
- oracle.ias.cache.Shutdown
- oracle.ias.cache.Startup
- oracle.security.jps.wls.JpsWlsStartupClass
- oracle.jrf.wls.JRFStartup
- oracle.core.ojdl.weblogic.ODLConfiguration
- oracle.bpel.services.common.util.GenerateBPMCryptoKey
- oracle.j2ee.ws.server.WebServiceServerStartup

B.2 Applications and Resources for the WebCenter Content Managed Servers

The following applications and resources are targeted to the Oracle WebCenter Content cluster, wcc_cluster, which includes these Managed Servers:

- WLS_WCC1 on WCCHOST1
- WLS_WCC2 on WCCHOST2

Applications for WebCenter Content

- DMS Application (11.1.1.1.0)
- Oracle UCM Help
- Oracle UCM Native Web Service
- Oracle UCM Web Services
- Oracle WebCenter Content - Content Server
- wsil-wls

Data Sources for WebCenter Content

- CSDS
- leasing

Startup and Shutdown Classes for WebCenter Content

- oracle.jrf.AppContextStartup
- oracle.dms.wls.DMSStartup
- oracle.dms.wls.DMSShutdown
- oracle.as.jmx.framework.wls.spi.StartupListener
- oracle.ias.cache.Shutdown
- oracle.ias.cache.Startup
- oracle.security.jps.wls.JpsWlsStartupClass
- oracle.jrf.wls.JRFStartup
- oracle.core.ojdl.weblogic.ODLConfiguration
- oracle.j2ee.ws.server.WebServiceServerStartup

B.3 Applications and Resources for the Inbound Refinery Managed Servers

The following applications and resources are targeted to the Oracle WebCenter Content: Inbound Refinery cluster, ibr_cluster, which includes these Managed Servers:

- WLS_IBR1 on WCCHOST1
- WLS_IBR2 on WCCHOST2

Applications for Inbound Refinery

- DMS Application (11.1.1.1.0)
- Oracle UCM Help
- Oracle UCM Native Web Services
- Oracle UCM Web Services
- Oracle WebCenter Content - Inbound Refinery
- wsil-wls

Startup and Shutdown Classes for Inbound Refinery

- `oracle.jrf.AppContextStartup`
- `oracle.dms.wls.DMSStartup`
- `oracle.dms.wls.DMSShutdown`
- `oracle.as.jmx.framework.wls.spi.StartupListener`
- `oracle.ias.cache.Shutdown`
- `oracle.ias.cache.Startup`
- `oracle.security.jps.wls.JpsWlsStartupClass`
- `oracle.jrf.wls.JRFStartup`
- `oracle.core.ojdl.weblogic.ODLConfiguration`
- `oracle.j2ee.ws.server.WebServiceServerStartup`

B.4 Applications and Resources for the Imaging Managed Servers

The following applications and resources are targeted to the Oracle WebCenter Content: Imaging cluster, img_cluster, which includes these Managed Servers:

- WLS_IMG1 on WCCHOST1
- WLS_IMG2 on WCCHOST2

Applications for Imaging

- DMS Application (11.1.1.1.0)
- imaging
- imaging-vc
- wsil-wls

Data Sources for Imaging

- IPMDS
- mds-owsm

Persistence Stores for Imaging

- IMGJMSServer1Store
- IMGJMSServer2Store

JMS Deployment Modules for Imaging

- IpmJmsServer1
- IpmJmsServer2
- ViewerJmsServer1
- ViewerJmsServer3

Startup and Shutdown Classes for Imaging

- oracle.jrf.AppContextStartup
- oracle.dms.wls.DMSStartup
- oracle.dms.wls.DMSShutdown
- oracle.as.jmx.framework.wls.spi.StartupListener
- oracle.ias.cache.Shutdown
- oracle.ias.cache.Startup
- oracle.security.jps.wls.JpsWlsStartupClass
- oracle.jrf.wls.JRFStartup
- oracle.core.ojdl.weblogic.ODLConfiguration
- oracle.j2ee.ws.server.WebServiceServerStartup

B.5 Applications and Resources for the Administration Server

The following applications and resources are targeted to the Oracle WebLogic Server Administration Server, AdminServer, which runs on SOAHOST1 and SOAHOST2.

Applications for the Administration Server

- DMS Application (11.1.1.1.0)
- em
- FMW Welcome Page Application (11.1.0.0.0)
- NonJ2EEManagement (11.1.1)
- wsil-wls

Data Sources for the Administration Server

- mds-owsm
- mds-soa

Startup and Shutdown Classes for the Administration Server

- oracle.jrf.AppContextStartup
- oracle.dms.wls.DMSStartup
- oracle.dms.wls.DMSShutdown
- oracle.as.jmx.framework.wls.spi.StartupListener
- oracle.ias.cache.Shutdown
- oracle.ias.cache.Startup
- oracle.security.jps.wls.JpsWlsStartupClass
- oracle.jrf.wls.JRFStartup
- oracle.core.ojdl.weblogic.ODLConfiguration
- oracle.j2ee.ws.server.WebServiceServerStartup

Index

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W

X