Oracle® Fusion Middleware

Enterprise Deployment Guide for Oracle Enterprise Content Management Suite

11*g* Release 1 (11.1.1)

E15483-04

July 2011

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

E15483-04

Copyright © 2010, 2011, Oracle and/or its affiliates. All rights reserved.

Primary Author: Ron van de Crommert

Contributors: Janga Aliminati (architect), Fermin Castro Alonso, Richard Delval, Pradeep Bhat, Eileen He, Van Sioung Ng Yan Tun, Madhwa Chintarevula, Izeta Delic, Nouar Garcia-mardambek, Pascal Prevot

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

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

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

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

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

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

This software and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

6 Extending the Domain with SOA Components

7 Extending the Domain with Oracle UCM

8 Extending the Domain with Oracle I/PM

9 Setting Up Node Manager

10 Configuring Server Migration

11 Integration with Oracle Identity Management

12 Managing the Topology

Index

Preface

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

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/accessibility/`.

Accessibility of Code Examples in Documentation

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

Accessibility of Links to External Web Sites in Documentation

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

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit `http://www.oracle.com/support/contact.html` or visit `http://www.oracle.com/accessibility/support.html` if you are hearing impaired.

Related Documents

For more information, see the following documents:

- *Oracle Fusion Middleware Quick Installation Guide for Oracle Enterprise Content Management Suite*
- *Oracle Fusion Middleware Installation Guide for Oracle Enterprise Content Management Suite*
- *Oracle Fusion Middleware Administrator's Guide*

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 section covers these topics:

1.1 What Is an Enterprise Deployment?

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

An Oracle Fusion Middleware enterprise deployment:

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

For more information on high availability practices, go to `http://www.oracle.com/technology/deploy/availability/htdocs/maa.htm`.

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

1.2 About Oracle Enterprise Content Management Suite

Oracle Enterprise Content Management Suite (Oracle ECM), an Oracle Fusion Middleware component, is an integrated suite of products designed for managing content. It is the industry's most unified enterprise content management platform that enables you to leverage industry-leading document management, Web content management, digital asset management, and records management functionality to build your business applications. Building a strategic enterprise content management infrastructure for content and applications helps you to reduce costs, easily share content across the enterprise, minimize risk, automate expensive, time-intensive and manual processes, and consolidate multiple Web sites onto a single platform.

Oracle Enterprise Content Management Suite offers the following benefits:

- Superior usability: Built-in support for end-users, workgroups, content experts, process owners, administrators and webmaster
- Optimized management: Unified architecture for securely managing documents, files, web content and digital assets
- Hot-pluggable: Out-of-the-box support for Oracle and third-party repositories, security systems, and enterprise applications

The reference enterprise deployment topology in this guide includes the following Oracle Enterprise Content Management Suite components:

- Oracle Universal Content Management
- Oracle Imaging and Processing Management

Oracle Universal Content Management

Oracle Universal Content Management (Oracle UCM) is the industry's most unified enterprise content management platform that enables you to leverage market-leading document management, Web content management, digital asset management, and records retention functionality to build and complement your business applications. Building a strategic enterprise content management infrastructure for content and applications helps you to reduce costs, easily share content across the enterprise, minimize risk, automate expensive, time-intensive and manual processes, and consolidate multiple Web sites onto a single platform for centralized management. Through user-friendly interfaces, roles-based authentication and security models, Oracle Universal Content Management empowers users throughout the enterprise to view, collaborate on or retire content, ensuring that all accessible distributed or published information is secure, accurate and up-to-date.

Oracle Imaging and Processing Management

Oracle Imaging and Process Management (Oracle I/PM) provides organizations with a scalable solution focused on process-oriented imaging applications and image-enabling enterprise applications. It enables image capture, annotation and markup of images, automates routing and approvals, and supports high-volume applications for billions of items. With Oracle Imaging and Process Management, organizations can quickly integrate their content and processes directly with Oracle and other third-party enterprise applications.

1.3 Benefits of Oracle Recommendations

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

- Section 1.3.1, "Built-In Security"
- Section 1.3.2, "High Availability"

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

1.3.1 Built-In Security

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

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

> **Note:** The Oracle Technology Network (`http://www.oracle.com/technology/index.html`) provides a list of validated load balancers and their configuration at `http://www.oracle.com/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 (OID) is isolated in the data tier.
- Oracle Identity Management (IDM) components are in a separate subnet.
- All communication between components across protection zones is restricted by port and protocol, according to firewall rules.

1.3.2 High Availability

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

1.4 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.
- **WebLogic Server home**: A WebLogic Server home contains installed files necessary to host a WebLogic Server. The WebLogic Server home directory is a peer of Oracle home directories and resides within the directory structure of the Middleware home.
- **Middleware home:** A Middleware home consists of the Oracle WebLogic Server home, and, optionally, one or more Oracle homes. A Middleware home can reside on a local file system or on a remote shared disk that is accessible through NFS.
- **Oracle instance**: An Oracle instance contains one or more active middleware system components, for example Oracle Web Cache, Oracle HTTP Server, or Oracle Internet Directory. You determine which components are part of an instance, either at install time or by creating and configuring an instance at a later time. An Oracle instance contains files that can be updated, such as configuration files, log files, temporary files.
- **failover**: When a member of a high availability system fails unexpectedly (unplanned downtime), in order to continue offering services to its consumers, the system 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 *Oracle Fusion Middleware High Availability Guide* for information on active-active and active-passive systems.
- **failback**: After a system undergoes a successful failover operation, the original failed member can be repaired over time and be re-introduced into the system as a standby member. If desired, a failback process can be initiated to activate this member and deactivate the other. This process reverts the system back to its pre-failure configuration.
- **hardware cluster**: A hardware cluster is a collection of computers that provides a single view of network services (for example: an IP address) or application services (for example: databases, Web servers) to clients of these services. Each node in a hardware cluster is a standalone server that runs its own processes. These processes can communicate with one another to form what looks like a single system that cooperatively provides applications, system resources, and data to users.

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

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

  Managed server homes can also be optionally located in the shared disk. The shared storage can be a Network Attached Storage (NAS), a Storage Area Network (SAN) or any other storage system that multiple nodes can access simultaneously and can read/write.
- **primary node**: The node that is actively running an Oracle Fusion Middleware instance at any given time and has been configured to have a backup/secondary node. If the primary node fails, Oracle Fusion Middleware instance is failed over to the secondary node. This failover can be manual or automated using the Clusterware for Administration Server. For a server migration based scenario, WebLogic Whole Server Migration is used for automated failover.
- **secondary node**: The node that is the backup node for an Oracle Fusion Middleware instance. This is where the active instance fails over when the primary node is no longer available. See the definition for primary node in this section.
- **network host name**: Network host name is a name assigned to an IP address either through the `/etc/hosts` file or through DNS resolution. This name is visible in the network that the machine to which it refers to is connected. Often, the network host name and physical host name are identical. However, each machine has only one physical host name but may have multiple network host names. Thus, a machine's network host name may not always be its physical host name.
- **physical host name**: This guide differentiates between the terms physical host name and network host name. This guide uses physical host name to refer to the "internal name" of the current machine. On UNIX, this is the name returned by the `hostname` command.

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

- **physical IP**: Physical IP refers to the IP address of a machine on the network. In 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**: Virtual host name is a network addressable host name that maps to one or more physical machines via a load balancer or a hardware cluster. For load balancers, the name "virtual server name" is used interchangeably with virtual host name in this book. A load balancer can hold a virtual host name on behalf of a set of servers, and clients communicate indirectly with the machines using the virtual host name. A virtual host name in a hardware cluster is a network host name assigned to a cluster virtual IP. Because the cluster virtual IP is not permanently attached to any particular node of a cluster, the virtual host name is not permanently attached to any particular node either.

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

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

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

  A load balancer also uses a virtual IP as the entry point to a set of servers. These servers tend to be active at the same time. This virtual IP address is not assigned to any individual server but to the load balancer which acts as a proxy between servers and their clients.

1.5 Abbreviations

The following abbreviations are used in this enterprise deployment guide:

- ASM = Automatic Storage Management
- BPEL = Business Process Execution Language
- BPM = Business Process Management
- CSF = Credential Store Framework
- DB = Database
- DN = Distinguished Name
- DNS = Domain Name Service
- ECM = Enterprise Content Management
- EM = Enterprise Manager
- FMW = Fusion Middleware
- FTP = File Transfer Protocol
- HA = High Availability
- IDM = Identity Management
- IP = Internet Protocol
- I/PM = Imaging and Processing Management
- JOC = Java Object Cache
- JRF = Java Required Files
- JMS = Java Message Service
- LBR = Load Balancing Router
- LDAP = Lightweight Directory Access Protocol
- NAS = Network Attached Storage
- NAT = Network Address Translation
- NFS = Network File System
- OAM = Oracle Access Manager
- OAP = Oracle Access Protocol
- OCS = Oracle Content Server
- OHS = Oracle HTTP Server
- OID = Oracle Internet Directory
- OIP = Oracle Internet Protocol
- OPSS = Oracle Platform Security Services
- OVD = Oracle Virtual Directory
- RAC = Real Application Clusters
- RCA = Repository Create Assistant
- RCU = Repository Creation Utility
- SAN = Storage Area Network
- SLA = Service Level Agreement
- SOA = Service-Oriented Architecture
- UCM = Universal Content Management
- UMS = Unified Messaging System

- URM = Universal Records Management
- VIP = Virtual Internet Protocol
- WC = WebCenter
- WLS = WebLogic Server
- WLST = WebLogic Scripting Tool
- WSM = Web Services Manager

1.6 Hardware Requirements

Typical hardware requirements for the enterprise deployment on Linux operating systems are listed in Table 1–1. For detailed requirements or for requirements for other platforms, see the Oracle Fusion Middleware installation documentation for that platform.

Table 1–1 Typical Hardware Requirements

Server	Processor	Disk	Memory	TMP Directory	Swap
Database	4 or more X Pentium, 1.5 GHz or greater	*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*	4 or more X Pentium, 1.5 GHz or greater	10 GB	4 GB	Default	Default
SOAHOST*n*	4 or more X Pentium, 1.5 GHz or greater	10 GB[1]	4 GB	Default	Default
ECMHOST*n*	4 or more X Pentium, 1.5 GHz or greater	10 GB[2]	6 GB	Default	Default

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

[2] ECM can reuse MW_HOME binaries from the SOA installation in shared storage.

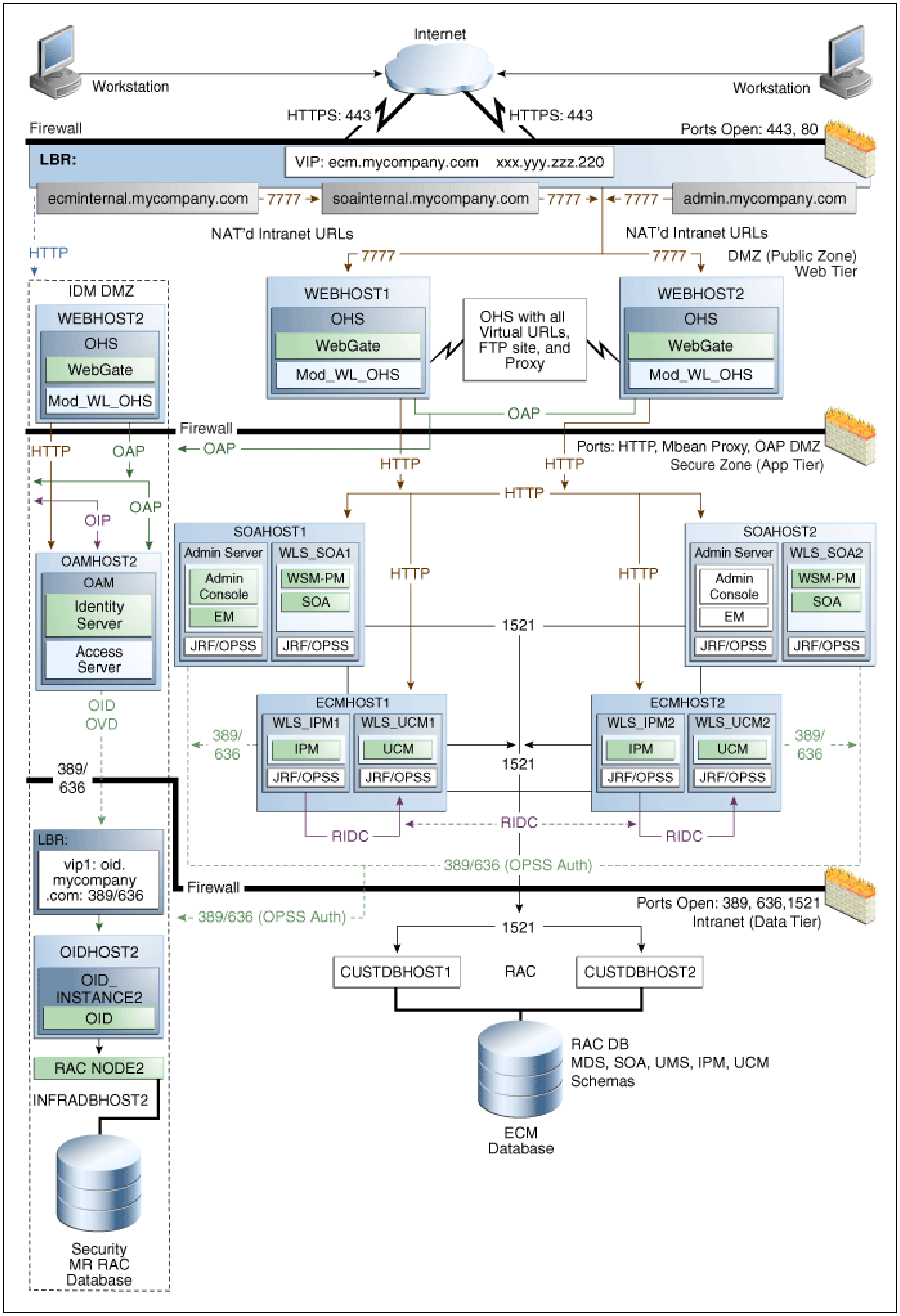
Note: You must perform the appropriate capacity planning to determine the number of nodes, CPU, and memory requirements for each node depending on the specific system's load as well as the throughput and response requirements. These will vary for each application or custom SOA system being used.

1.7 Enterprise Deployment Reference Topology

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

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

Figure 1–1 Reference Topology for Oracle ECM Enterprise Deployment



This section covers these topics:

- Section 1.7.1, "Oracle Identity Management"
- Section 1.7.2, "Web Tier"
- Section 1.7.3, "Application Tier"
- Section 1.7.4, "Data Tier"
- Section 1.7.5, "Unicast Requirement"

1.7.1 Oracle Identity Management

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

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

1.7.2 Web Tier

Nodes in the web tier are located in the DMZ public zone. In this tier, two nodes WEBHOST1 and WEBHOST2 run Oracle HTTP Server configured with WebGate and mod_wl_ohs.

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

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

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

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

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

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 EDG.

1.7.3 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 SOA components such as BPEL Process Manager. The managed servers are configured in an active-active manner.

ECMHOST1 and ECMHOST2 run the Oracle Universal Content Management (UCM) servers and Oracle Imaging and Processing Management (I/PM) 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 5.12, "Manually Failing Over the Administration Server to SOAHOST2"). Alternatively, you can configure the Oracle WebLogic Server Administration Console with CFC/CRS to fail over automatically on a separate hardware cluster (not shown in this architecture).

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

1.7.4 Data Tier

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

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

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

1.7.5 Unicast Requirement

Oracle recommends that the nodes in the Oracle ECM 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.

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

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

1.8 What to Install

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

Table 1–2 Components and Installation Sources

Component	Distribution Medium
Oracle Database 10*g* or 11*g*	Oracle Database 10*g* CD (10.2.0.4 or later SE or EE version of the database) using the AL32UTF8 character set. Oracle Database Server 11*g* CD (11.1.0.7 or later SE or EE version of the database), using the AL32UTF8 character set.
Repository Creation Utility (RCU)	Oracle Fusion Middleware Repository Creation Utility 11*g* (11.1.1.5) DVD
Oracle WebLogic Server (WLS)	Oracle WebLogic Server (10.3.5) DVD
Oracle HTTP Server (OHS)	Oracle Fusion Middleware WebTier and Utilities 11*g* (11.1.1.5) DVD
Oracle SOA Suite	Oracle SOA Suite 11*g* (11.1.1.5) DVD
Oracle Enterprise Content Management Suite	Oracle ECM 11*g* (11.1.1.5) DVD
Oracle Access Manager (OAM) Webgate	Oracle Access Manager 10*g* Webgates (10.1.4.3) DVD ; OAM OHS 11*g* webgates per platform
Oracle Virtual Directory (OVD)	Oracle Identity and Access Management 11*g* (11.1.1.5) DVD
Oracle Internet Directory (OID)	Oracle Identity and Access Management 11*g* (11.1.1.5) DVD

1.9 Installation Procedure

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

The Oracle Fusion Middleware Configuration Wizard enables you to extend the Oracle WebLogic Server domain by adding only the needed components. Rather than using the Configuration Wizard to create SOA components and the Oracle Enterprise Content Management Suite components along with the domain that includes the Administration Server, Enterprise Manager, and WSM-PM in a single pass, you can instead create the domain and its Administration Server in one pass of the Configuration Wizard and then extend the domain by adding Oracle Enterprise Manager, BPM Suite, SOA, WSM-PM, Oracle Universal Content Management (UCM), and Oracle Imaging and Process Management (I/PM) components in subsequent passes. Using this incremental approach, you can verify the installation of the servers and perform specific validations after each pass of the Configuration Wizard. In general, Oracle recommends the following approach:

1. Run a first pass of the Configuration Wizard to install the Administration Server and Oracle Enterprise Manager (described in Chapter 5, "Creating a Domain with Administration Server").
2. Run a second pass of the Configuration Wizard to install the SOA and WSM-PM components (described in Section 6, "Extending the Domain with SOA Components").
3. Run a third pass of the Configuration Wizard to install the Oracle UCM components (described in Chapter 7, "Extending the Domain with Oracle UCM").
4. Run a fourth pass of the Configuration Wizard to install the Oracle I/PM components (described in Chapter 8, "Extending the Domain with Oracle I/PM").

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

Some variation from the topology in Section 1.7, "Enterprise Deployment Reference Topology" is possible. For example, if a deployment chooses to install Oracle UCM alone, then only sections applicable to Oracle UCM need to be followed. Also, in this case, it is expected that the Administration Server will exist on ECMHOST1 instead and the instructions on creating the domain should be modified appropriately.

2

Database and Environment Preconfiguration

This chapter describes database and network environment preconfiguration required by the Oracle ECM enterprise deployment topology, as well as recommendations for shared storage and directory structure. It contains the following sections:

- Section 2.1, "Database"
- Section 2.2, "Network"
- Section 2.3, "Shared Storage and Recommended Directory Structure"
- Section 2.4, "LDAP as Credential and Policy Store"

2.1 Database

You must install the Oracle Fusion Middleware repository before you can configure the Oracle Fusion Middleware components. You install the Oracle Fusion Middleware metadata repository into an existing database using the Repository Creation Utility (RCU), which is available from the RCU DVD or from the location listed in Table 1–2.

For the enterprise topology, an Oracle Real Application Clusters (RAC) database is highly recommended. When you configure the Oracle ECM components, the Oracle Fusion Middleware Configuration Wizard will prompt you to enter the information for connecting to the database that contains the metadata repository.

This section covers these topics:

- Section 2.1.1, "Setting Up the Database"
- Section 2.1.2, "Loading the Oracle Fusion Middleware Metadata Repository in the Oracle RAC Database"
- Section 2.1.3, "Backing Up the Database"

2.1.1 Setting Up the Database

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

- Section 2.1.1.1, "Database Host Requirements"
- Section 2.1.1.2, "Supported Database Versions"
- Section 2.1.1.3, "Initialization Parameters"
- Section 2.1.1.4, "Database Services"

2.1.1.1 Database Host Requirements

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

- **Oracle Clusterware**

  For 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 *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.

2.1.1.2 Supported Database Versions

Oracle Enterprise Content Management Suite 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/technology/software/products/ias/files/fusion_certification.html

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

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

> **Note:** Oracle ECM 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.

2.1.1.3 Initialization Parameters

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

Table 2–1 Required Initialization Parameters

Configuration	Parameter	Required Value	Parameter Class
SOA	`PROCESSES`	400 or greater	Static
ECM	`PROCESSES`	100 or greater	Static
SOA and ECM	`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:

```
SQL> SHOW PARAMETER processes
```

Set the initialization parameter using the following command:

```
SQL> 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. See the *Oracle Database Administrator's Guide* for details on parameter files, server parameter files, and how to change parameter values.

2.1.1.4 Database Services

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

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

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

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

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

   ```
   prompt> srvctl add service -d ecmdb -s ecmedg -r ecmdb1,ecmdb2
   ```

3. Start the service using `srvctl`:

   ```
   prompt> srvctl start service -d ecmdb -s ecmedg
   ```

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

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

> **Note:** For simplicity, the datasource configuration screens in this guide use the same service name (ecmedg.mycompany.com)

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

Perform these steps to load the Oracle Fusion Middleware Repository into a database:

1. Insert the Repository Creation Utility (RCU) DVD, 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**.

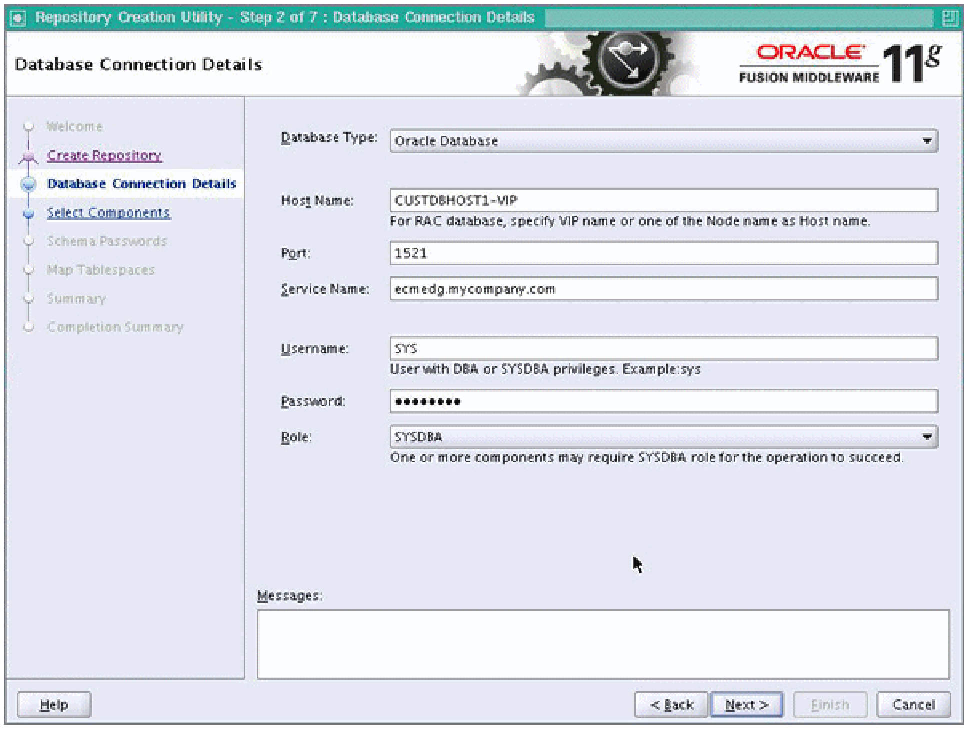
3. In the Create Repository screen, select **Create** to load component schemas into a database. Click **Next**.

4. In the Database Connection Details screen, enter 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: CUSTDBHOST1-VIP.
   - **Port**: Specify the listen port number for the database: 1521.
   - **Service Name**: Specify the service name of the database (ecmedg.mycompany.com).
   - **Username**: Specify the name of the user with DBA or SYSDBA privileges: `SYS`.
   - **Password**: Enter the password for the SYS user.
   - **Role**: Select the database user's role from the list: `SYSDBA` (required by the SYS user).

   Click **Next**.

Figure 2–1 Database Connection Details Screen



Repository Creation Utility - Step 2 of 7 : Database Connection Details

Database Connection Details

Database Type: Oracle Database

Host Name: CUSTDBHOST1-VIP

For RAC database, specify VIP name or one of the Node name as Host name.

Port: 1521

Service Name: ecmedg.mycompany.com

Username: SYS

User with DBA or SYSDBA privileges. Example:sys

Password: ••••••••

Role: SYSDBA

One or more components may require SYSDBA role for the operation to succeed.

Messages:

5. In the Select Components screen, 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 *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**

     – SOA and BPM Infrastructure:

     - **SOA Infrastructure**

     - **User Messaging**

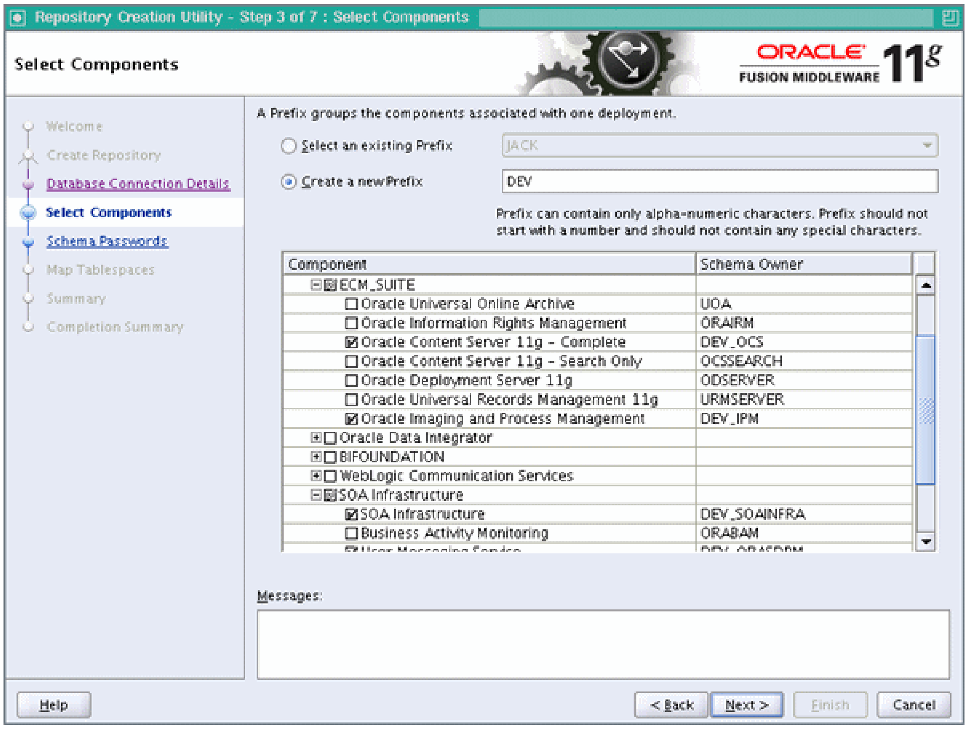
     – Enterprise Content Management:

     - **Oracle Content Server 11g - Complete**

     - **Oracle Imaging and Process Management**

   Click **Next**.

Figure 2–2 Select Components Screen



Repository Creation Utility - Step 3 of 7 : Select Components

Select Components

- Welcome
- Create Repository
- Database Connection Details
- Select Components
- Schema Passwords
- Map Tablespaces
- Summary
- Completion Summary

A Prefix groups the components associated with one deployment.

○ Select an existing Prefix: JACK

● Create a new Prefix: DEV

Prefix can contain only alpha-numeric characters. Prefix should not start with a number and should not contain any special characters.

Component	Schema Owner
⊟☒ ECM_SUITE	
☐ Oracle Universal Online Archive	UOA
☐ Oracle Information Rights Management	ORAIRM
☑ Oracle Content Server 11g - Complete	DEV_OCS
☐ Oracle Content Server 11g - Search Only	OCSSEARCH
☐ Oracle Deployment Server 11g	ODSERVER
☐ Oracle Universal Records Management 11g	URMSERVER
☑ Oracle Imaging and Process Management	DEV_IPM
⊞☐ Oracle Data Integrator	
⊞☐ BIFOUNDATION	
⊞☐ WebLogic Communication Services	
⊟☒ SOA Infrastructure	
☑ SOA Infrastructure	DEV_SOAINFRA
☐ Business Activity Monitoring	ORABAM
☑ User Messaging Service	DEV_ORASDPM

Messages:

6. In the Schema Passwords screen, enter passwords for the main and additional (auxiliary) schema users, and click **Next**.

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

7. In the Map Tablespaces screen, choose the tablespaces for the selected components, and click **Next**.
8. In the Summary screen, click **Create**.
9. In the Completion Summary screen, click **Close**.

Note: Oracle recommends using the database used for identity management (see Chapter 11, "Integration with Oracle Identity Management") to store the Oracle WSM policies. It is therefore expected to use the IM database information for the OWSM MDS schemas, which will be different from the one used for the rest of SOA schemas. To create the required schemas in the database, repeat the steps above using the IM database information, but select only "AS Common Schemas: Metadata Services" in the Select Components screen (step 5).

2.1.3 Backing Up the Database

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

Backing up the database is for the explicit purpose of quick recovery from any issue that may occur in the further steps. You can choose to use your backup strategy for the database for this purpose or simply 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.

2.2 Network

This section covers these topics:

- Section 2.2.1, "Virtual Server Names"
- Section 2.2.2, "Load Balancers"
- Section 2.2.3, "IPs and Virtual IPs"
- Section 2.2.4, "Firewalls and Ports"

2.2.1 Virtual Server Names

The Oracle ECM enterprise topology uses the following virtual server names:

- Section 2.2.1.1, "ecm.mycompany.com"
- Section 2.2.1.2, "admin.mycompany.com"
- Section 2.2.1.3, "soainternal.mycompany.com"
- Section 2.2.1.4, "ecminternal.mycompany.com"

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

2.2.1.1 ecm.mycompany.com

`ecm.mycompany.com` is a virtual server name that acts as the access point for all HTTP traffic to the run-time Oracle ECM components. Traffic to SSL is configured. Clients access this service using the address `ecm.mycompany.com:443`. This virtual server is defined on the load balancer.

2.2.1.2 admin.mycompany.com

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

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

This virtual server is defined on the load balancer.

2.2.1.3 soainternal.mycompany.com

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

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

This virtual server is defined on the load balancer.

2.2.1.4 ecminternal.mycompany.com

`ecminternal.mycompany.com` is a virtual server name used for internal invocations of ECM services. This URL is not exposed to the internet and is only accessible from the intranet.

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

This virtual server is defined on the load balancer.

2.2.2 Load Balancers

This enterprise topology uses an external load balancer. For more information on load balancers, see Section 1.7.2, "Web Tier."

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

Configuring the Load Balancer

Perform these steps to configure the load balancer:

1. Create a pool of servers. You will assign this pool to virtual servers.
2. Add the addresses of the Oracle HTTP Server hosts to the pool. For example:
   - WEBHOST1:7777
   - WEBHOST2:7777
3. Configure a virtual server in the load balancer for soainternal.mycompany.com:80.
   - For this virtual server, use your internal SOA address as the virtual server address (for example, `soainternal.mycompany.com`). This address is typically not externalized.
   - Specify HTTP as the protocol.
   - Enable address and port translation.
   - Enable reset of connections when services and/or nodes are down.
   - Assign the pool created in step 1 to the virtual server.
4. Configure a virtual server in the load balancer for `ecm.mycompany.com:443`.
   - For this virtual server, use your system's frontend address as the virtual server address (for example, `ecm.mycompany.com`). The frontend address is the externally facing host name used by your system and that will be exposed in the Internet.
   - Configure this virtual server with port 80 and port 443. Any request that goes to port 80 should be redirected to port 443.
   - Specify HTTP as the protocol.
   - Enable address and port translation.
   - Enable reset of connections when services and/or nodes are down.
   - Assign the pool created in step 1 to the virtual server.
   - 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`.
   - For this virtual server, use your internal administration address as the virtual server address (for example, `admin.mycompany.com`). This address is typically not externalized.
   - Specify HTTP as the protocol.
   - Enable address and port translation.
   - Enable reset of connections when services and/or nodes are down.
   - 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 a virtual server in the load balancer for `ecminternal.mycompany.com:80`.
   - For this virtual server, use your internal ECM address as the virtual server address (for example, `ecminternal.mycompany.com`). This address is typically not externalized.

- Specify HTTP as the protocol.
- Enable address and port translation.
- Enable reset of connections when services and/or nodes are down.
- Assign the pool created in step 1 to the virtual server.
- Optionally, create rules to filter out access to `/console` and `/em` on this virtual server.

7. Configure monitors for the Oracle HTTP Server nodes to detect failures in these nodes.

   - Set up a monitor to regularly ping the "`/`" URL context.

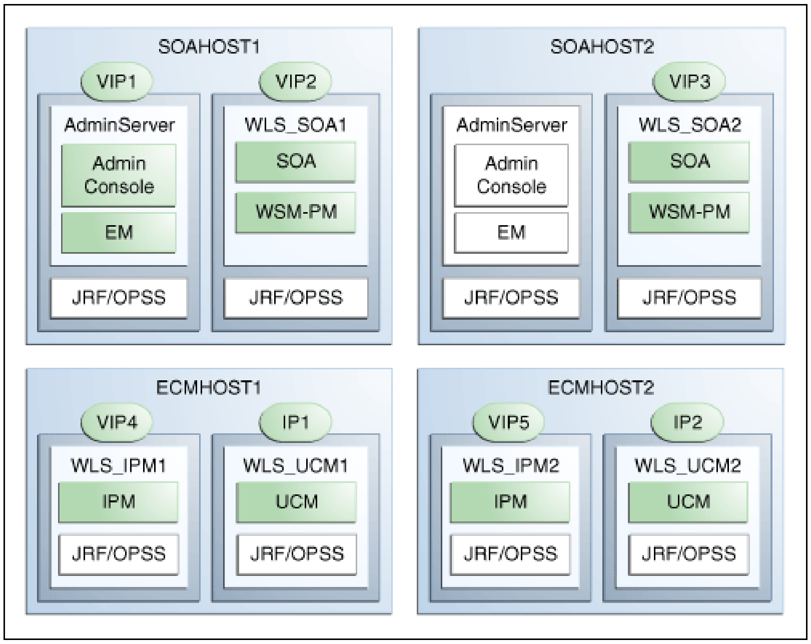
     Tip: Use `GET /\n\n` instead if the Oracle HTTP Server's document root does not include `index.htm` and Oracle WebLogic Server returns a 404 error for "`/`".

   - 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 SOA system, that is, specify a value greater than the longest period of time any of your requests to HTTP servers can take.

2.2.3 IPs and Virtual IPs

Configure the Administration Server and the managed servers to listen on different virtual IPs and physical IPs as illustrated in Figure 2–3.

Figure 2–3 IPs and VIPs Mapped to Administration Server and Managed Servers



As shown in Figure 2–3, each VIP and IP is attached to the Oracle WebLogic server that uses it. VIP1 is failed manually to restart the Administration Server in SOAHOST2. VIP2 and VIP3 fail over from SOAHOST1 to SOAHOST2 and from

SOAHOST2 to SOAHOST1, respectively, through the Oracle WebLogic Server migration feature. WLS_IPM1 and WLS_IPM2 also use server migration to fail over VIP4 and VIP5, respectively, from ECMHOST1 to ECMHOST2. See the *Oracle Fusion Middleware High Availability Guide* for information on the WebLogic Server Migration feature. Physical (non-virtual) IPs are fixed to each node. IP1 is the physical IP of ECMHOST1 and is used as the listen address by the WLS_UCM1 server. IP2 is the physical IP of ECMHOST2 and is used as the listen address by the WLS_UCM2 server.

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

Table 2–2 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	ECMHOST1VHN1	ECMHOST1VHN1 is the virtual host name that maps to the listen address for WLS_IPM1 and fails over with server migration of this managed server. It is enabled on the node where WLS_IPM1 process is running (ECMHOST1 by default).
VIP5	ECMHOST2VHN1	ECMHOST2VHN1 is the virtual host name that maps to the listen address for WLS_IPM2 and fails over with server migration of this managed server. It is enabled on the node where WLS_IPM2 process is running (ECMHOST2 by default).

2.2.4 Firewalls and Ports

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

Most port numbers are assigned during installation.

Table 2–3 lists the ports used in the Oracle ECM topology, including the ports that you must open on the firewalls in the topology.

Firewall notation:

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

Table 2–3 Ports Used

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Browser request	FW0	80	HTTP / Load Balancer	Inbound	Timeout depends on all HTML content and the type of process model used for SOA.
Browser request	FW0	443	HTTPS / Load Balancer	Inbound	Timeout depends on all HTML content and the type of process model used for SOA.
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 SOA.
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 SOA.
Callbacks and outbound invocations	FW1	80	HTTP / Load Balancer	Outbound	Timeout depends on all HTML content and the type of process model used for SOA.
Callbacks and Outbound invocations	FW1	443	HTTPS / Load Balancer	Outbound	Timeout depends on all HTML content and the type of process model used for SOA.
Load balancer to Oracle HTTP Server	n/a	7777	HTTP	n/a	See Section 2.2.2, "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).
SOA and WSM server access	FW1	8001 Range: 8000 - 8080	HTTP / WLS_SOA*n*	Inbound	Timeout varies based on the type of process model used for SOA.
UCM access	FW1	16200	HTTP / WLS_UCM*n*	Inbound	Browser-based access. Configurable session timeouts.
I/PM access	FW1	16000	HTTP / WLS_IPM*n*	Inbound	Browser-based access. Configurable session timeouts.
I/PM connection to UCM	n/a	4444	HTTP / WLS_IPM*n*	Inbound	Persistent connection. Timeout configurable on UCM 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.

Table 2–3 (Cont.) Ports Used

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Communication between UCM 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 IPM 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	HTTP / Administration Server and Enterprise Manager t3	Both	You should tune this timeout based on the type of access to the administration console (whether it is planned to use the Oracle WebLogic Server Administration Console from application tier clients or clients external to the application tier).
Node Manager	n/a	5556	TCP/IP	n/a	n/a For actual values, see "Firewalls and Ports" in *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*.
Access Server access	FW1	6021	OAP	Inbound	For actual values, see "Firewalls and Ports" in Oracle Fusion *Middleware Enterprise Deployment Guide for Oracle Identity Management*.
Identity Server access	FW1	6022	OAP	Inbound	
Database access	FW2	1521	SQL*Net	Both	Timeout depends on all database content and on the type of process model used for SOA.
Coherence for deployment	n/a	8088 Range: 8000 - 8090		n/a	n/a

Table 2–3 (Cont.) Ports Used

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Oracle Internet Directory access	FW2	389	LDAP	Inbound	You should tune the directory server's parameters based on load balancer, and not the other way around.
Oracle Internet Directory access	FW2	636	LDAP SSL	Inbound	You should tune the directory server's parameters based on load balancer, and not the other way around.
JOC for OWSM	n/a	9991 Range: 9988-9998	TCP/IP	n/a	n/a

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

2.3 Shared Storage and Recommended Directory Structure

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

This section covers these topics:

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

2.3.1 Terminology for Directories and Directory Environment Variables

This enterprise deployment guide uses the following references to directory locations:

- **ORACLE_BASE:** This environment variable and related directory path refers to the base directory under which Oracle products are installed.
- **MW_HOME:** This environment variable and related directory path refers to the location where Fusion Middleware (FMW) resides.
- **WL_HOME:** This environment variable and related directory path contains installed files necessary to host a WebLogic Server.
- **ORACLE_HOME:** This environment variable and related directory path refers to the location where Oracle Fusion Middleware SOA Suite or Oracle Enterprise Content Management Suite is installed.
- **ORACLE_COMMON_HOME:** This environment variable and related directory path refers to the Oracle home that contains the binary and library files required for the Oracle Enterprise Manager Fusion Middleware Control and Java Required Files (JRF).

- **Domain directory:** This directory path refers to the location where the Oracle WebLogic domain information (configuration artifacts) is stored. Different WLS Servers can use different domain directories even when in the same node.
- **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 updatable files, such as configuration files, log files, and temporary files.

> **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 in Linux to refer to /u01/app/oracle (that is, the recommended ORACLE_BASE location). In Windows, you would use %ORACLE_BASE% and use Windows-specific commands.

2.3.2 Recommended Locations for the Different Directories

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

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

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

You can use a shared domain directory for all managed servers in different nodes or use one domain directory per node. Sharing domain directories for managed servers facilitates the scale-out procedures. In this case, the deployment should conform to the requirements (if any) of the storage system to facilitate multiple machines mounting the same shared volume. The configuration steps provided in this enterprise deployment topology assume that a local (per node) domain directory is used for each managed server.

All procedures that apply to multiple local domains apply to a single shared domain. Hence, this enterprise deployment guide uses a model where one domain directory is used per node. The directory can be local or reside in shared storage. JMS file stores and JTA transaction logs need to be placed on a shared storage in order to ensure that they are available from multiple boxes for recovery in the case of a server failure or migration.

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

ORACLE_BASE:

/u01/app/oracle

MW_HOME (application tier):

ORACLE_BASE/product/fmw

- Mount point: *ORACLE_BASE*/product/fmw
- Shared storage location: *ORACLE_BASE*/product/fmw (VOL1 and VOL2)
- 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 EDG for ECM, SOAHOST1 and ECMHOST1 mount VOL1 and SOAHOST2 and ECMHOST2 mount VOL2. When only one volume is available, nodes mount two different directories in shared storage alternatively (that is, for example, SOAHOST1 would use *ORACLE_BASE*/product/fmw1 as shared storage location and SOAHOST2 would use *ORACLE_BASE*/product/fmw2 as 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 MW_HOMEs would be available; at least one at *ORACLE_BASE*/product/fmw1, and another at *ORACLE_BASE*/product/fmw2. These MW_HOMEs are mounted on the same mount point in all nodes.

MW_HOME (web tier):

ORACLE_BASE/product/fmw/web

- Mount point: *ORACLE_BASE*/product/fmw
- Shared storage location: *ORACLE_BASE*/product/fmw (VOL1 and VOL2)
- 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 EDG for ECM, 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 shared storage location and WEBHOST2 would use *ORACLE_BASE*/product/fmw2 as shared storage location).

> **Note:** Web tier installation is usually performed on local storage 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*/ecm

ORACLE_COMMON_HOME:

MW_HOME/oracle_common

ORACLE_INSTANCE:

ORACLE_BASE/admin/*instance_name*

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

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

Domain Directory for Administration Server Domain Directory:

ORACLE_BASE/admin/*domain_name*/aserver/*domain_name* (The last "*domain_name*' is added by 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*/Node*n*/mserver/ (each node uses a different domain directory for managed servers).

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

Location for JMS file-based stores and Tlogs:

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 SOA and ECM 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.

Location for Oracle I/PM input files, images, and samples input directories:

ORACLE_BASE/admin/*domain_name*/*ipm_cluster_name*/input_files

ORACLE_BASE/admin/*domain_name*/*ipm_cluster_name*/input_files/Samples

ORACLE_BASE/admin/*domain_name*/*ipm_cluster_name*/images

- Mount point: *ORACLE_BASE*/admin/*domain_name*/*ipm_cluster_name*
- Shared storage location: *ORACLE_BASE*/admin/*domain_name*/*ipm_cluster_name*
- Mounted from: All nodes containing I/PM 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 installation of Oracle I/PM, 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:** In order to process input files, the input agent must have the appropriate permissions on 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 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 for Oracle UCM's vault (native file repository):

ORACLE_BASE/admin/*domain_name*/*ucm_cluster_name*/cs/vault

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

Location for application directory for 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

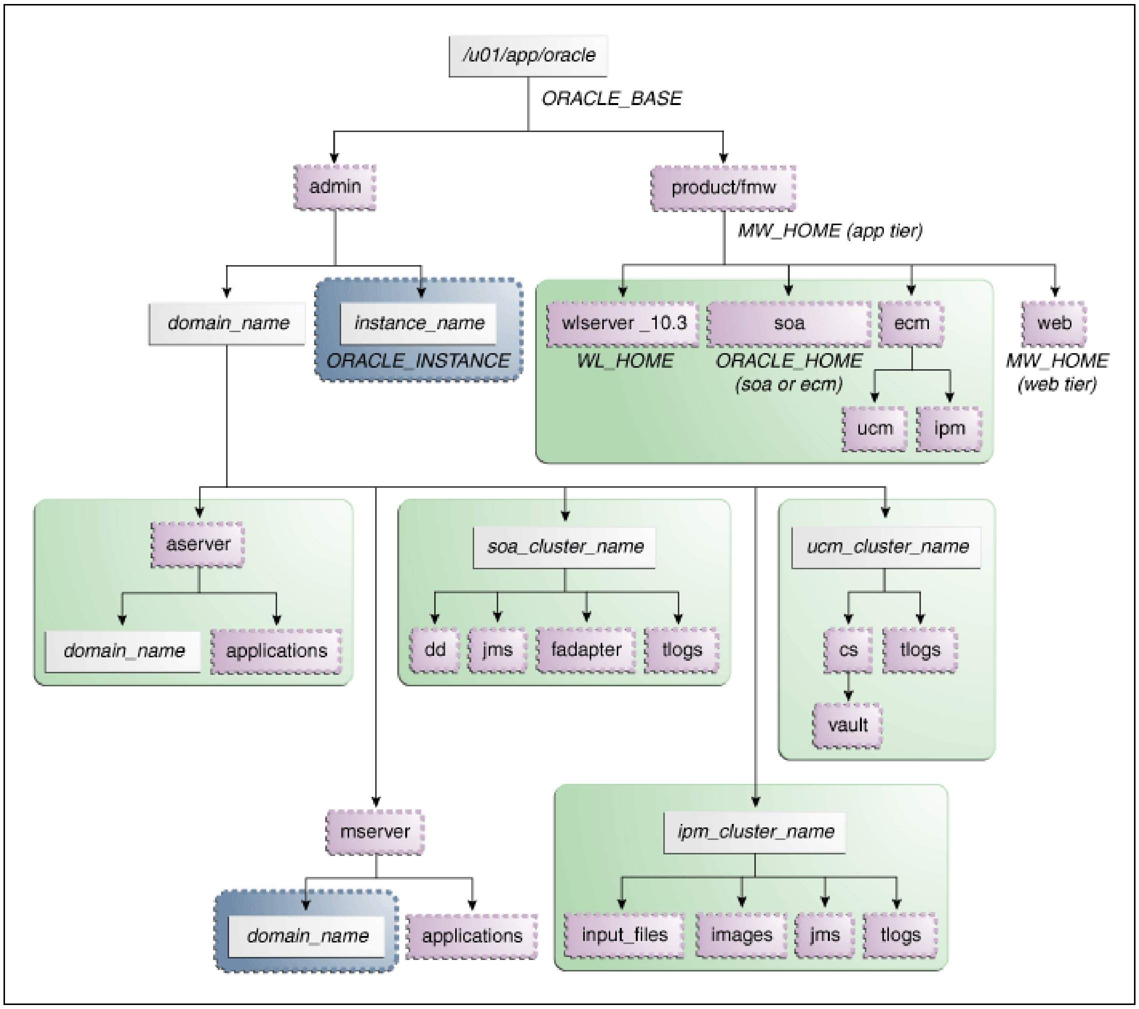
Location for application directory for managed server:

ORACLE_BASE/admin/*domain_name*/mserver/applications

> **Note:** This directory is local in the context of the EDG for ECM.

Figure 2–4 shows this directory structure in a diagram.

Figure 2–4 EDG Directory Structure for Oracle ECM



The directory structure in Figure 2–4 does not show other required internal directories such as `oracle_common` and `jrockit`.

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

Table 2–4 Directory Structure Elements

Element	Explanation
🟩	The administration server domain directories, applications, deployment plans, file adapter control directory, JMS and TX logs, and the entire MW_HOME are on a shared disk.
🟦	The managed server domain directories can be on a local disk or a shared disk. Further, if you want to share the managed server domain directories on multiple nodes, then you must mount the same shared disk location across the nodes. The *instance_name* directory for the web tier can be on a local disk or a shared disk.
🟪	Fixed name.
⬜	Installation-dependent name.

Figure 2–5 shows an example configuration for shared storage with multiple volumes for SOA and ECM.

Figure 2–5 Example Configuration for Shared Storage

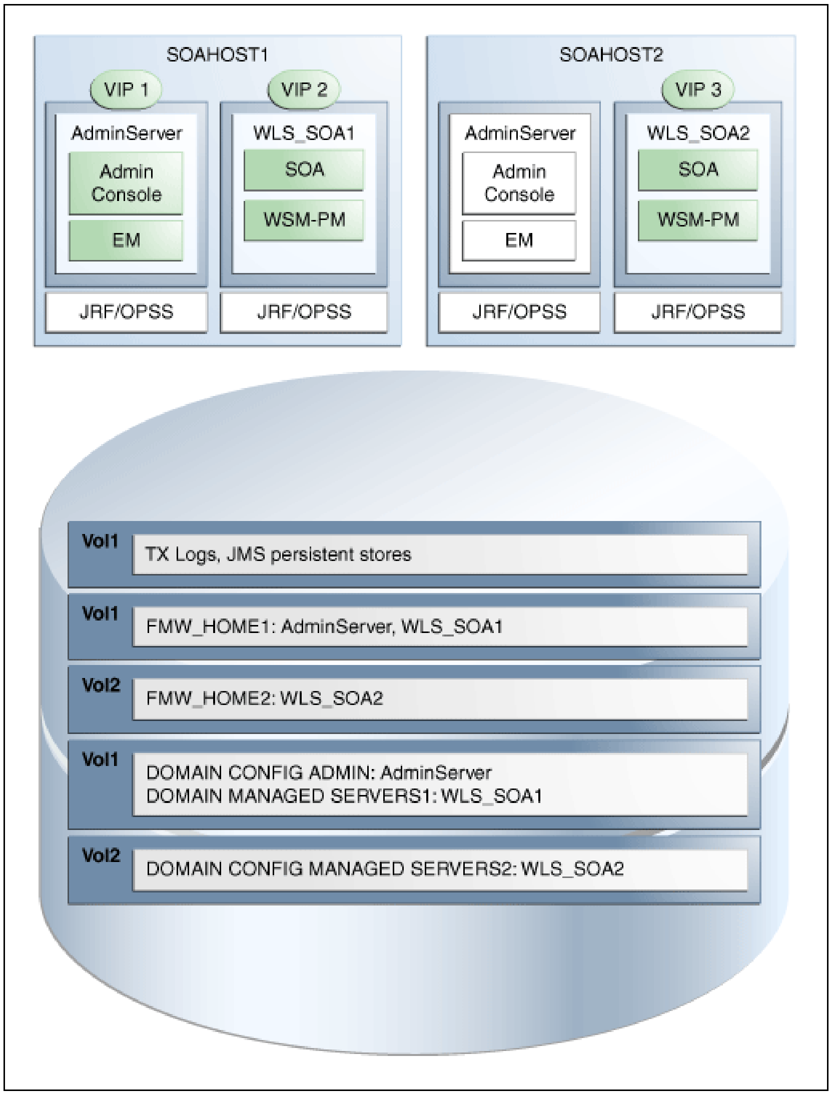


Table 2–5 summarizes the directory structure for the domain.

Table 2–5 Contents of Shared Storage

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

2.3.3 Shared Storage Configuration

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

"nasfiler" is the shared storage filer.

From SOAHOST1:

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

From SOAHOST2:

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

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

From SOAHOST1:

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

From SOAHOST2:

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

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

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

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

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

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

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

2.4 LDAP as Credential and Policy Store

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

An Oracle ECM enterprise deployment topology uses different domain homes for the administration server and the managed server as described in the Section 2.3, "Shared Storage and Recommended Directory Structure." Derived from this, and for integrity and consistency purposes, Oracle requires the use of an LDAP server as policy and credential store in context of an Oracle ECM enterprise deployment topology. Follow the steps in Section 11.1.2.1, "Creating the LDAP Authenticator" to configure the Oracle ECM enterprise deployment with an LDAP server as credential and policy store.

3

Installing the Software

This chapter describes the software installations required for the enterprise deployment reference topology for Oracle Enterprise Content Management Suite. The installation is divided into two parts. The first part covers the required web tier installations, while the second part addresses the required Fusion Middleware (FMW) components. Later chapters describe the required configuration steps to create the reference topology for Oracle Enterprise Content Management Suite.

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

This chapter contains the following sections:

- Section 3.1, "Software Installation Summary"
- Section 3.2, "Installing Oracle HTTP Server"
- Section 3.3, "Installing Oracle Fusion Middleware"
- Section 3.4, "Applying Patchsets to Oracle Fusion Middleware Components"

3.1 Software Installation Summary

Table 3–1 shows what software should be installed on each host or be accessible from each host.

Table 3–1 Software To Be Installed On Each Host or Accessible From Each Host

Hosts	OHS	WLS	SOA	ECM
WEBHOST1	X			
WEBHOST2	X			
SOAHOST1		X	X	
SOAHOST2		X	X	
ECMHOST1		X		X
ECMHOST2		X		X

Table 3–2 shows the software versions used.

Table 3–2 Software Versions Used

Software	Name	Version
OHS	Oracle HTTP Server 11*g*	11.1.1.2 + Patchset 11.1.1.5 (PS4)
WLS	WebLogic Server 11*g*	10.3.5
SOA	Oracle SOA Suite 11*g*	11.1.1.5
ECM	Oracle Enterprise Content Management 11*g*	11.1.1.5
IDM	Oracle Identity Management 11*g*	11.1.1.2 + Patchset 11.1.1.5 (PS4)

3.2 Installing Oracle HTTP Server

This section covers these topics:

- Section 3.2.1, "Installing Oracle HTTP Server on WEBHOST1 and WEBHOST2"
- Section 3.2.2, "Backing Up the Installation"

3.2.1 Installing Oracle HTTP Server on WEBHOST1 and WEBHOST2

Prerequisites

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 installation documentation.
- 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

Procedure

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

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 screen to execute /createCentralInventory.sh as root.

   Click **OK**.

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

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

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

6. 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**.

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

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

3.2.2 Backing Up the Installation

The Fusion Middleware Home should be backed up now (make sure no server is running at this point):

```
WEBHOST1> tar -cvpf fmwhomeback.tar ORACLE_BASE/product/fmw HOME/oraInventory
```

3.3 Installing Oracle Fusion Middleware

This section describes how to install the required Oracle Fusion Middleware software for the enterprise deployment reference topology for Oracle Enterprise Content Management Suite. The software components to be installed consist of the Oracle WebLogic Server Home (WL_HOME) and Oracle Home (ORACLE_HOME). As described in Section 2.3, "Shared Storage and Recommended Directory Structure," you install Oracle Fusion Middleware in at least two storage locations for redundancy.

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

This section covers these topics:

- Section 3.3.1, "Installing Oracle WebLogic Server and Creating the Fusion Middleware Home"
- Section 3.3.2, "Installing Oracle Fusion Middleware Components"
- Section 3.3.3, "Backing Up the Installation"

3.3.1 Installing Oracle WebLogic Server and Creating the Fusion Middleware Home

Perform these steps 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 from the installation media:

   ```
   $ ./wls1034_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**.

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

   Click **Next**.
4. In the Register for Security Updates screen, enter your contact information so that you can be notified of security updates, and click **Next**.
5. In the Choose Install Type screen, select **Custom**, and click **Next**.
6. In the Choose Products and Components screen, click **Next**.
7. In the JDK Selection screen, select *only* **Oracle JRockit 1.6.0_<*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_3.6**, 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** check box and click **Done**.

3.3.2 Installing Oracle Fusion Middleware Components

This section covers these topics:

- Section 3.3.2.1, "Installing Oracle Fusion Middleware SOA Suite"
- Section 3.3.2.2, "Installing Oracle Fusion Middleware ECM Suite"

3.3.2.1 Installing Oracle Fusion Middleware SOA Suite

Perform these steps to install Oracle Fusion Middleware SOA Suite on SOAHOST1 and SOAHOST2:

> **Note:** Since the installation is performed on a shared storage, the MW_HOME is accessible and used by the Oracle ECM servers in ECMHOST1 and ECMHOST2.

1. On Linux platforms, if the /etc/oraInst.loc file exists, check that its contents are correct. Specifically, check that the inventory directory is correct and that you have write permissions for that directory. If the /etc/oraInst.loc file does not exist, you can skip this step.
2. Start the installer for Oracle Fusion Middleware SOA Suite from the installation media:

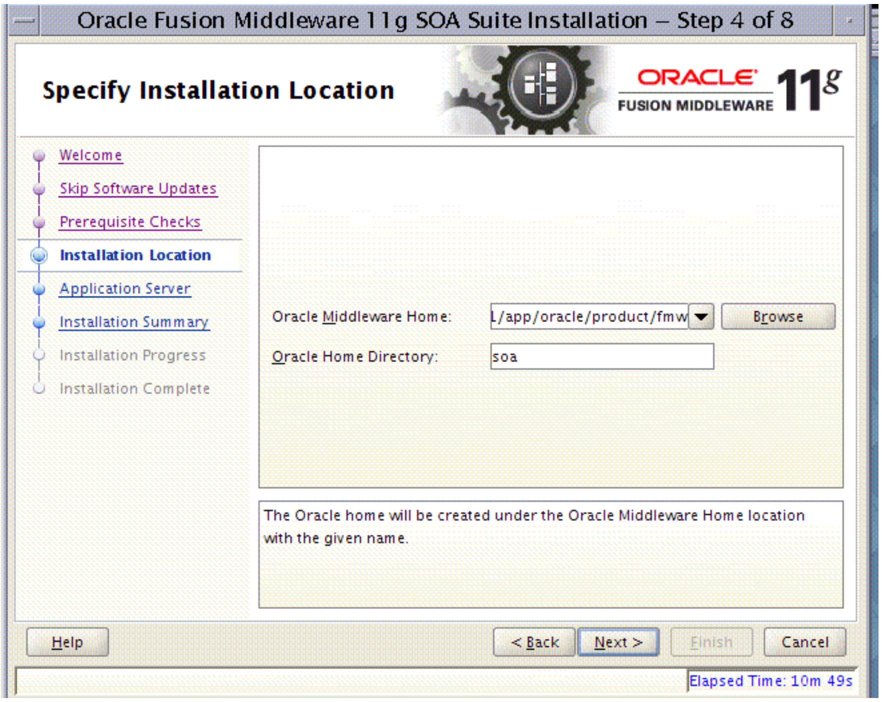
   ```
   ./runInstaller
   ```

   When the installer prompts you for a JRE/JDK location, enter the Oracle SDK location created in the Oracle WebLogic Server installation (see Section 3.3.1, "Installing Oracle WebLogic Server and Creating the Fusion Middleware Home"; for example, ***ORACLE_BASE*****/product/fmw/jdk160_<*****version*****>**).
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, choose Skip Software Updates and click **Next**.
6. In the Prerequisite Checks screen, verify that all checks complete successfully, and click **OK**.
7. In the Specify Installation Location screen, provide the installation location for Oracle Fusion Middleware SOA Suite. Select the previously installed Oracle Middleware Home from the drop-down list. For the Oracle Home directory, enter the directory name (**soa**).

Figure 3–1 Specify Installation Location Screen in Installer Wizard



Oracle Fusion Middleware 11g SOA Suite Installation – Step 4 of 8

Specify Installation Location

Oracle Middleware Home: l/app/oracle/product/fmw

Oracle Home Directory: soa

The Oracle home will be created under the Oracle Middleware Home location with the given name.

Elapsed Time: 10m 49s

Click **Next** when you are done.

8. In the Application Server screen, select **WebLogic Server** and click **Next**.

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

   The Oracle Fusion Middleware SOA Suite software is installed.

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

3.3.2.2 Installing Oracle Fusion Middleware ECM Suite

When you install Oracle Fusion Middleware Enterprise Content Management Suite, you install the product bits for the following applications on your system:

- Oracle Universal Content Management (Oracle UCM)
- Oracle Imaging and Process Management (Oracle I/PM)
- Oracle Information Rights Management (Oracle IRM)
- Oracle Universal Records Management (Oracle URM)
- Oracle Inbound Refinery (Oracle IBR)

Perform these steps to install Oracle Enterprise Content Management Suite on SOAHOST1 and SOAHOST2:

> **Note:** Since the installation is performed on a shared storage, the MW_HOME is accessible and used by the Oracle ECM servers in ECMHOST1 and ECMHOST2.

1. Start the installer for Oracle Enterprise Content Management 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 (see Section 3.3.1, "Installing Oracle WebLogic Server and Creating the Fusion Middleware Home"; for example, ***ORACLE_BASE*/product/fmw/jdk160_<*version*>**).

2. In the Welcome screen, click **Next**.
3. In the Install Software Updates screen, choose 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 Enterprise Content Management Suite. Select the previously installed Oracle Middleware Home from the drop-down list. For the Oracle Home directory, enter the directory name (**ecm**).

   Click **Next** when you are done.
6. In the Installation Summary screen, click **Install**.

   The Oracle Enterprise Content Management Suite software is installed.
7. In the Installation Complete screen, click **Finish**.

3.3.3 Backing Up the Installation

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

```
SOAHOST1> 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.

3.4 Applying Patchsets to Oracle Fusion Middleware Components

This enterprise deployment guide assumes that the binary installations for both Fusion Middleware components and Oracle HTTP Server are patched to the latest patchset available. The EDG does not take into consideration running systems and rolling patches, but instead assumes the system is created from scratch and patched to the latest available patchset before the Oracle Fusion Middleware Configuration Wizard and domain operations are started. See the documentation included with each patchset to properly patch an existing installation.

Specifically for the Enterprise Deployment Guide for Oracle Enterprise Content Management Suite, it is expected that the SOA installation used by Oracle I/PM is patched to the Oracle Fusion Middleware 11.1.1.5 (PS4) patchset level and that the Oracle WebLogic Server installation is using version 10.3.5 (PS4).

4 Configuring the Web Tier

This chapter describes how to configure the Oracle Web Tier to support the Oracle Fusion Middleware SOA Suite implementation.

This chapter contains the following sections:

- Section 4.1, "Configuring the Oracle Web Tier"
- Section 4.2, "Validating the Installation"
- Section 4.3, "Associating the Oracle Web Tier with the Oracle WebLogic Domain"
- Section 4.4, "Configuring Oracle HTTP Server with the Load Balancer"
- Section 4.5, "Configuring Virtual Hosts"

4.1 Configuring the Oracle Web Tier

Prior to configuration, the Oracle web tier software must be installed on WEBHOST1 and WEBHOST2, as described in Section 3.2, "Installing Oracle HTTP Server." The steps for configuring the Oracle web tier are the same for both WEBHOST1 and WEBHOST2.

Perform these steps to configure the Oracle web tier:

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

   ```
   WEBHOST1> cd ORACLE_HOME/bin
   ```

2. Start the Configuration Wizard:

   ```
   WEBHOST1> ./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: /u01/app/oracle/admin/web*n*
   - AS Instance Name: web*n*
   - OHS Component Name: ohs*n*

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

   Click **Next**.

6. In high-availability implementations, whilst not mandatory, it makes life simpler if all of the ports used by the various components are synchronized across hosts. Oracle allows automatic port configuration to be bypassed by specifying ports to be used in a file.

   In the Configure Ports screen, select a file name and then click **View/Edit**. 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 e-mail 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.

4.2 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/
```

4.3 Associating the Oracle Web Tier with the Oracle WebLogic Domain

Once an Oracle WebLogic domain has been created, the Oracle web tier can be linked to the domain. The advantages of doing this are that the Oracle web tier can be managed and monitored via the Oracle Fusion Middleware console.

To associate the Oracle web tier with the WebLogic domain, execute the following commands on both WEBHOST1 and WEBHOST2:

```
WEBHOSTn> cd ORACLE_BASE/admin/instance_name/bin
WEBHOSTn> ./opmnctl registerinstance -adminHost ADMINVHN  -adminPort 7001
-adminUsername weblogic
```

4.4 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.

4.5 Configuring Virtual Hosts

In order for Oracle Enterprise Content Management Suite to work with the load balancer, virtual hosts need to be created in the Oracle HTTP Server configuration. Edit the httpd.conf file located at *ORACLE_INSTANCE*/config/OHS/*component_name* and add the following to the virtual host section:

```
NameVirtualHost *:7777
<VirtualHost *:7777>
   ServerName https://ecm.mycompany.com:443
   RewriteEngine On
   RewriteOptions inherit
   UseCanonicalName On
</VirtualHost>

NameVirtualHost *:7777
<VirtualHost *:7777>
   ServerName admin.mycompany.com:80
   RewriteEngine On
   RewriteOptions inherit
   UseCanonicalName On
</VirtualHost>

NameVirtualHost *:7777
<VirtualHost *:7777>
   ServerName soainternal.mycompany.com:80
   RewriteEngine On
   RewriteOptions inherit
   UseCanonicalName On
</VirtualHost>
```

5 Creating a Domain with Administration Server

This chapter describes how to create a domain using the Oracle Fusion Middleware Configuration Wizard, Oracle WebLogic Server Administration Console, and Oracle Enterprise Manager. You can extend the domain to add Fusion Middleware (FMW) components: SOA, Oracle Universal Content Management (UCM) and, optionally, Oracle Imaging and Process Management (I/PM). This will be addressed in later chapters in this document.

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

This chapter contains the following sections:

5.1 Enabling ADMINVHN 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 will associate the Administration Server with a virtual IP (ADMINVHN). Make sure that ADMINVHN is enabled on SOAHOST1.

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

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

For example:

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

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

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

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

```
/bin/ping 100.200.140.206
```

> **Note:** In these examples, 'eth*X*' is the ethernet interface (eth0 or eth1) and *Y* is the index (0, 1, 2, etc.).

5.2 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. Change the directory to the location of the Oracle Fusion Middleware Configuration Wizard (created in Chapter 3, "Installing the Software"):

   ```
   SOAHOST1> cd ORACLE_COMMON_HOME/common/bin
   ```

2. Start the Configuration Wizard:

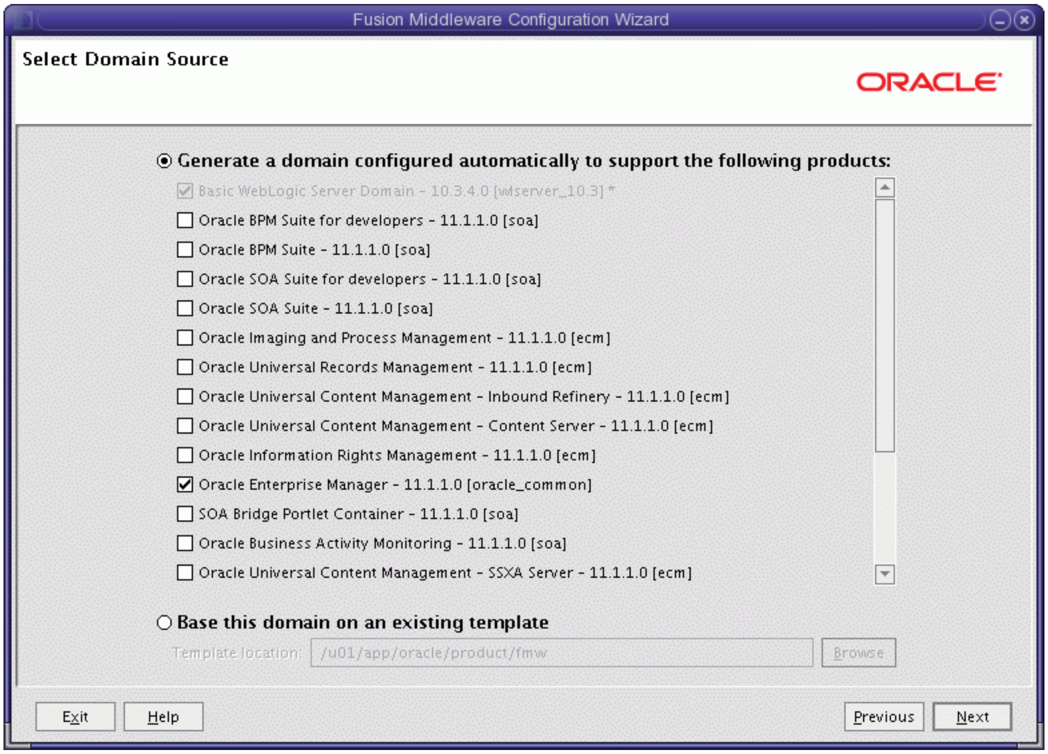
   ```
   SOAHOST1> ./config.sh
   ```

3. In the Welcome screen, select **Create a new WebLogic Domain**, and click **Next**.

4. The Select Domain Source screen opens. In this screen, do the following (as shown in Figure 5–1):

   - Select **Generate a domain configured automatically to support the following products**.
   - Select the following products:
     - **Basic WebLogic Server Domain - 10.3.5.0 [wlserver_10.3]** (this should be selected automatically)
     - **Oracle Enterprise Manager - 11.1.1.0 [oracle_common]**
     - **Oracle JRF - 11.1.1.0 [oracle_common]** (this should be selected automatically)

Figure 5–1 Select Domain Source Screen



Select Domain Source

● Generate a domain configured automatically to support the following products:

☑ Basic WebLogic Server Domain - 10.3.4.0 [wlserver_10.3] *
☐ Oracle BPM Suite for developers - 11.1.1.0 [soa]
☐ Oracle BPM Suite - 11.1.1.0 [soa]
☐ Oracle SOA Suite for developers - 11.1.1.0 [soa]
☐ Oracle SOA Suite - 11.1.1.0 [soa]
☐ Oracle Imaging and Process Management - 11.1.1.0 [ecm]
☐ Oracle Universal Records Management - 11.1.1.0 [ecm]
☐ Oracle Universal Content Management - Inbound Refinery - 11.1.1.0 [ecm]
☐ Oracle Universal Content Management - Content Server - 11.1.1.0 [ecm]
☐ Oracle Information Rights Management - 11.1.1.0 [ecm]
☑ Oracle Enterprise Manager - 11.1.1.0 [oracle_common]
☐ SOA Bridge Portlet Container - 11.1.1.0 [soa]
☐ Oracle Business Activity Monitoring - 11.1.1.0 [soa]
☐ Oracle Universal Content Management - SSXA Server - 11.1.1.0 [ecm]

○ Base this domain on an existing template

Template location: /u01/app/oracle/product/fmw

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

- Oracle Enterprise Manager
- Oracle JRF

Click **Next**.

5. 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 2, "Database and Environment Preconfiguration": 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**.

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

   Click **Next**.

7. 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**.

8. In the Select Optional Configuration screen, select the following:

   - **Administration Server**
   - **Managed Servers, Clusters and Machines**

   Click **Next**.

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

   - Name: **AdminServer**
   - Listen address: enter **ADMINVHN**.
   - Listen port: **7001**
   - SSL listen port: **N/A**
   - SSL enabled: leave this check box unselected.

   Click **Next**.

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

11. In the Configure Clusters screen, click **Next**.

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

Table 5–1 Machines

Name	Node Manager Listen Address
ADMINVHN	localhost

   Leave all other fields to their default values. Please note that 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.

   Click **Next**.

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

   - **ADMINVHN:**
     - AdminServer

   Click **Next**.

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

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

5.3 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 username and password.

1. Create the following directory structure:

```
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 in the file:

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

   > **Note:** When you start the Administration Server, the username and password entries in the file get encrypted. You start the Administration Server in Section 5.5, "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.

5.4 Starting Node Manager on SOAHOST1

Perform these steps 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:

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

   > **Note:** You must use the `StartScriptEnabled` property to avoid class loading failures and other problems. See also Section 12.8.3, "Incomplete Policy Migration After Failed Restart of SOA Server."

2. Start Node Manager:

   ```
   SOAHOST1> cd WL_HOME/server/bin
   SOAHOST1> export JAVA_OPTIONS=-DDomainRegistrationEnabled=true
   SOAHOST1> ./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
   >
   > ```
   > SOAHOST1> ./startNodeManager.sh
   > ```
   >
   > (without the `export` command).

5.5 Starting the Administration Server on SOAHOST1

The administration server will be started and stopped using Node Manager. However, the first start of the administration server with Node Manager requires changing the default username 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. Start the Administration Server using the start script in the domain directory:

   ```
   SOAHOST1> cd ORACLE_BASE/admin/domain_name/aserver/domain_name/bin
   SOAHOST1> ./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 on ***domain_name***, then **Security**, then **General**, and then expand the **Advanced** options at the bottom.

   d. Click **Lock & Edit**.

   e. Enter a new username 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 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) and connect to Node Manager with `nmconnect` and the credentials set above, and start the administration server using `nmstart`:

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

   Once in the WLST shell, execute the following command (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 username 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.

5.6 Validating the Administration Server

Perform these steps to ensure 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 Oracle Enterprise Manager at http://ADMINVHN:7001/em.

4. Log in to Enterprise Manager Console with the username and password you specified in Section 5.3, "Creating boot.properties for the Administration Server on SOAHOST1."

5.7 Disabling Host Name Verification for the Administration Server

This step is required if you have not set up the appropriate certificates to authenticate the different nodes with the Administration Server (see Chapter 9, "Setting Up Node Manager"). If you have not configured the server certificates, you will receive errors when managing the different WebLogic servers. To avoid these errors, disable host name verification while setting up and validating the topology, and enable it again once the EDG topology configuration is complete as described in Chapter 9, "Setting Up Node Manager."

Perform these steps to disable host name verification:

1. Log in to Oracle WebLogic Server Administration Console.
2. Click **Lock & Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page opens.
5. Select **AdminServer(admin)** in the Names column of the table. The settings page for the server opens.
6. Open the SSL tab.
7. Expand the **Advanced** section of the page.
8. Set host name verification to 'None'.
9. Click **Save**.
10. Save and activate the changes.
11. The change will not take effect until the Administration Server is restarted (Node Manager must be up and running):

    a. Stop the Administration Server using the following command:

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

    b. Start the Administration Server again as described in Section 5.5, "Starting the Administration Server on SOAHOST1."

5.8 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 file *ORACLE_INSTANCE*/config/OHS/component/mod_wl_ohs.conf:

    ```
    # 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. For each of the web servers on WEBHOST1 and WEBHOST2, make sure the file *ORACLE_INSTANCE*/config/OHS/component/httpd.conf includes the following lines:

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

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

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

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

Important Security Consideration

For security purposes, and since the load balancer terminates SSL request (Oracle HTTP Server routes the requests as non-SSL to WebLogic Server), once SSL is configured for the load balancer, turn on the WebLogic plug-in enabled flag2 for the domain. Make sure you do this only if SSL is configured for the load balancer. To perform this procedure, follow these steps:

1. Log on to the Administration Console.
2. Click on 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** check box.
6. Save and activate the changes.
7. Restart the administration server (even though the WLS console may not specifically prompt for that).

5.9 Registering Oracle HTTP Server with WebLogic Server

For Oracle Enterprise Manager Fusion Middleware Console to be able to manage and monitor Oracle HTTP Server instances, they must be registered with the domain. To do this, you must register Oracle HTTP Server with Oracle WebLogic Server using the following command:

```
WEBHOST1> cd ORACLE_BASE/admin/instance_name/bin
WEBHOST1> ./opmnctl registerinstance -adminHost ADMINVHN -adminPort 7001
-adminUsername weblogic
```

You must also run this command from WEBHOST2 for OHS2.

> **Note:** After registering Oracle HTTP Server, it should appear as a manageable target in the Oracle Enterprise Manager Console. To verify this, log in to the Enterprise Manager Console. The WebTier item in the navigation tree should show that Oracle HTTP Server has been registered.

5.10 Setting the Frontend URL for the Administration Console

The Oracle 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. When the Administration Console is accessed using a load balancing router (LBR), it is required to change the Administration Server's frontend URL so that the user's web browser is redirected to the appropriate LBR address. To do this, complete these steps:

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

To eliminate redirections it is recommended that you disable the Administration Console's "Follow changes" feature. To do this, log on to the Administration Console and click **Preferences** and then **Shared Preferences**. Clear the 'Follow Configuration Changes' check box and click **Save**.

5.11 Validating Access Through Oracle HTTP Server

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 12.8, "Troubleshooting" for possible causes.

Validate Administration Console and Enterprise Manager through both Oracle HTTP Server instances using the following URLs:

- http://WEBHOST1:7777/console
- http://WEBHOST2:7777/console
- http://WEBHOST1:7777/em
- http://WEBHOST2:7777/em

> **Note:** After setting the frontend URL to the LBR address, the access to the console through the WEBHOST*n* addresses will be redirected by the console to the frontend URL, thus validating the correct configuration of both Oracle HTTP Server and the LBR device.

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

For information on configuring system access through the load balancer, see Section 2.2.2, "Load Balancers."

5.12 Manually Failing Over the Administration Server to SOAHOST2

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

- Section 5.12.1, "Assumptions and Procedure"
- Section 5.12.2, "Validating Access to SOAHOST2 Through Oracle HTTP Server"
- Section 5.12.3, "Failing the Administration Server Back to SOAHOST1"

5.12.1 Assumptions and Procedure

Please note the following assumptions:

- The Administration Server is configured to listen on ADMINVHN, and not on ANY address. See step 12 in Section 5.2, "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*, available in SOAHOST1 and SOAHOST2.

- The domain directory where the administration server is running in SOAHOST1 is on a shared storage and is mounted also from SOAHOST2.

> **Note:** Node Manager in SOAHOST2 will not control the domain *domain_name* at this point, since `unpack/nmEnroll` has not been run yet on SOAHOST2. However, for the purpose of AdminServer failover and control of the administration server itself, Node Manager will be fully functional. Later failovers (after `unpack` has been run in SOAHOST2) will correctly have Node Manager controlling the domain.

- Oracle WebLogic Server and Oracle FMW components have been installed in SOAHOST2 as described in Chapter 3, "Installing the Software" (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):
   ```
   SOAHOST1> /sbin/ifconfig ethX:Y down
   ```
   b. Run the following command on SOAHOST2:
   ```
   SOAHOST2> /sbin/ifconfig interface:index IP_address netmask netmask
   ```
   For example:
   ```
   /sbin/ifconfig eth0:1 100.200.140.206 netmask 255.255.255.0
   ```
   > **Note:** Make sure that the netmask and interface to be used match the available network configuration in SOAHOST2.

   c. Update the routing tables using `arping`, for example:
   ```
   SOAHOST2> /sbin/arping -b -A -c 3 -I eth0 100.200.140.206
   ```
3. Start Node Manager in SOAHOST2 as described in Section 5.4, "Starting Node Manager on SOAHOST1."
4. Start the Administration Server on SOAHOST2 as described in Section 5.5, "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 Oracle WebLogic Server Administration Console at http://ADMINVHN:7001/console.
   b. Check that you can access and verify the status of components in the Oracle Enterprise Manager at http://ADMINVHN:7001/em.

5.12.2 Validating Access to SOAHOST2 Through Oracle HTTP Server

Perform the same steps as in Section 5.11, "Validating Access Through Oracle HTTP Server." This is to check that you can access the Administration Server when it is running on SOAHOST2.

5.12.3 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.

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

3. Run the following command on SOAHOST1:

   ```
   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:

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

5. Start the Administration Server again on SOAHOST1 as described in Section 5.5, "Starting the Administration Server on SOAHOST1."
6. Test that you can access the Oracle WebLogic Server Administration Console at http://ADMINVHN:7001/console.
7. Check that you can access and verify the status of components in the Oracle Enterprise Manager at http://ADMINVHN:7001/em.

5.13 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. Also refer to the *Oracle Database Backup and Recovery Guide* for information on database backup.

Perform these steps to back up the installation at this point:

1. Back up the web tier:

   a. Shut down the instance using `opmnctl`:

   ```
   WEBHOST1> ORACLE_BASE/admin/instance_name/bin/opmnctl stopall
   ```

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

   ```
   WEBHOST1> tar -cvpf BACKUP_LOCATION/web.tar MW_HOME
   ```

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

   ```
   WEBHOST1> tar -cvpf BACKUP_LOCATION/web_instance_name.tar ORACLE_INSTANCE
   ```

   d. Start the instance using `opmnctl`:

   ```
   WEBHOST1> cd ORACLE_BASE/admin/instance_name/bin
   WEBHOST1> opmnctl startall
   ```

   Repeat this step for WEBHOST2.

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. 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 to create the backup:

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

   Do not forget to restart the Administration Server again.

6 Extending the Domain with SOA Components

This chapter describes how to extend a domain with Oracle WSM Policy Manager and SOA components using the Oracle Fusion Middleware Configuration Wizard. You can extend the resulting domain to add other Fusion Middleware components. 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 SOA components.

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

This chapter contains the following sections:

- Section 6.13, "Setting the Frontend HTTP Host and Port"
- Section 6.14, "Validating Access Through Oracle HTTP Server"
- Section 6.15, "Configuring a Shared JMS Persistence Store"
- Section 6.16, "Configuring a Default Persistence Store for Transaction Recovery"
- Section 6.17, "Enabling High Availability for Oracle File and FTP Adapters"
- Section 6.18, "Scaling the Oracle Database Adapter"
- Section 6.19, "Configuring Node Manager for the WLS_SOA Managed Servers"
- Section 6.20, "Configuring Server Migration for the WLS_SOA Managed Servers"
- Section 6.21, "Backing Up the Installation"

6.1 Backing Up the Installation

It is strongly recommended that you back up the existing Fusion Middleware Home and domain:

```
SOAHOST1> tar -cvpf fmwhomeback.tar ORACLE_BASE/product/fmw
SOAHOST1> tar -cvpf domainhomeback.tar ORACLE_BASE/admin/domain_name/aserver/
domain_name
```

This creates a backup of the installation files for both Oracle WebLogic Server and Oracle Fusion Middleware as well as the domain configuration.

6.2 Enabling SOAHOST1VHN1 on SOAHOST1 and SOAHOST2VHN1 on SOAHOST2

This step is required for server migration of WLS_SOA1 and WLS_SOA2. You will associate the WLS_SOA1 and WLS_SOA2 servers with virtual host names (SOAHOST1VHN1 and SOAHOST2VHN1). 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 (VIP2 and VIP3).

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

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

For example:

```
/sbin/ifconfig ethX:Y 100.200.140.205 netmask 255.255.255.0
```

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

```
/sbin/arping -q -U -c 3 -I ethX 100.200.140.205
```

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

```
/bin/ping 100.200.140.205
```

Note: In these examples, 'eth*X*' is the ethernet interface (eth0 or eth1) and *Y* is the index (0, 1, 2, etc.).

6.3 Running the Configuration Wizard on SOAHOST1 to Extend the Current Domain

Run the Oracle Fusion Middleware Configuration Wizard from the SOA home directory to extend a domain containing an Administration Server to support Oracle Web Services Manager and SOA components.

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

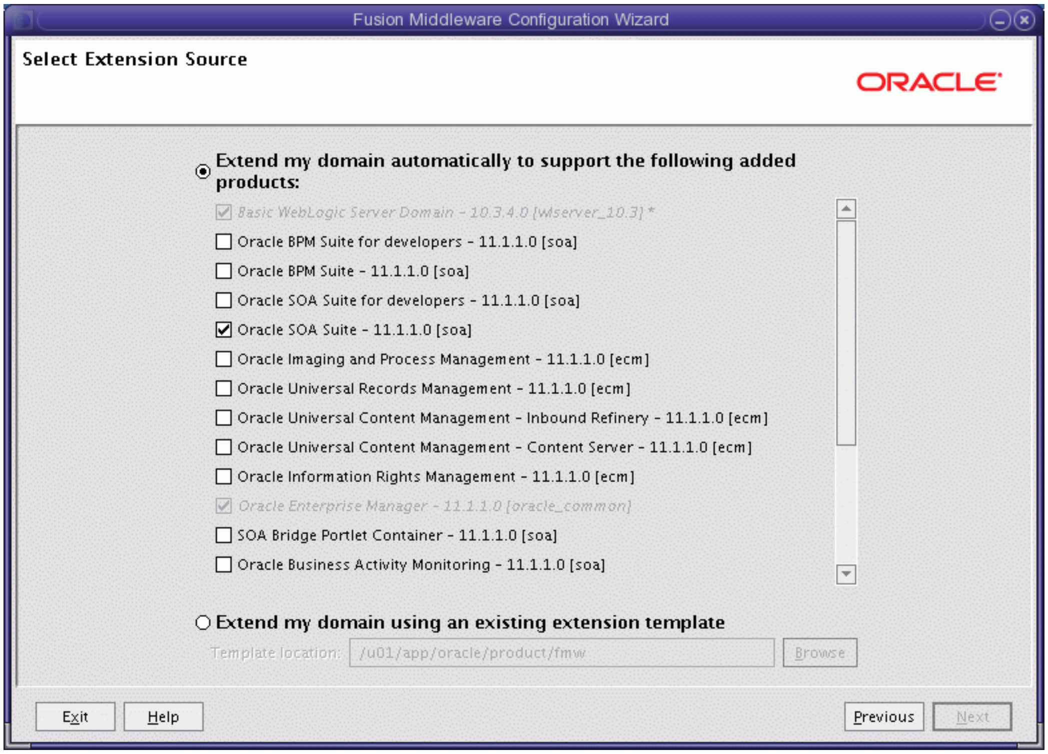
   ```
   SOAHOST1> cd ORACLE_COMMON_HOME/common/bin
   ```

3. Start the Configuration Wizard:

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

4. In the Welcome screen, select **Extend an Existing WebLogic Domain**, and click **Next**.
5. In the WebLogic Domain Directory screen, select the WebLogic domain directory (*ORACLE_BASE*/admin/*domain_name*/aserver/*domain_name*), and click **Next**.
6. The Select Extension Source screen opens. In this screen, do the following (as shown in Figure 6–1):
   - Select **Extend my domain automatically to support the following added products**.
   - Select the following products:
     - **Basic WebLogic Server Domain - 10.3.5.0 [wlserver_10.3]** (this should already be selected and grayed out)
     - **Oracle Enterprise Manager - 11.1.1.0 [oracle_common]** (this should already be selected and grayed out)
     - **Oracle SOA Suite - 11.1.1.0 [soa]**
     - **Oracle WSM Policy Manager - 11.1.1.0 [oracle_common]** (selected automatically when Oracle SOA suite is selected)
     - **Oracle JRF - 11.1.1.0 [oracle_common]** (this should already be selected and grayed out)

Figure 6–1 Select Extension Source screen for Oracle SOA



Select Extension Source

● Extend my domain automatically to support the following added products:

☑ Basic WebLogic Server Domain - 10.3.4.0 [wlserver_10.3] *
□ Oracle BPM Suite for developers - 11.1.1.0 [soa]
□ Oracle BPM Suite - 11.1.1.0 [soa]
□ Oracle SOA Suite for developers - 11.1.1.0 [soa]
☑ Oracle SOA Suite - 11.1.1.0 [soa]
□ Oracle Imaging and Process Management - 11.1.1.0 [ecm]
□ Oracle Universal Records Management - 11.1.1.0 [ecm]
□ Oracle Universal Content Management - Inbound Refinery - 11.1.1.0 [ecm]
□ Oracle Universal Content Management - Content Server - 11.1.1.0 [ecm]
□ Oracle Information Rights Management - 11.1.1.0 [ecm]
☑ Oracle Enterprise Manager - 11.1.1.0 [oracle_common]
□ SOA Bridge Portlet Container - 11.1.1.0 [soa]
□ Oracle Business Activity Monitoring - 11.1.1.0 [soa]

○ Extend my domain using an existing extension template

Template location: /u01/app/oracle/product/fmw

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

- Oracle SOA Suite
- Oracle WSM Policy Manager

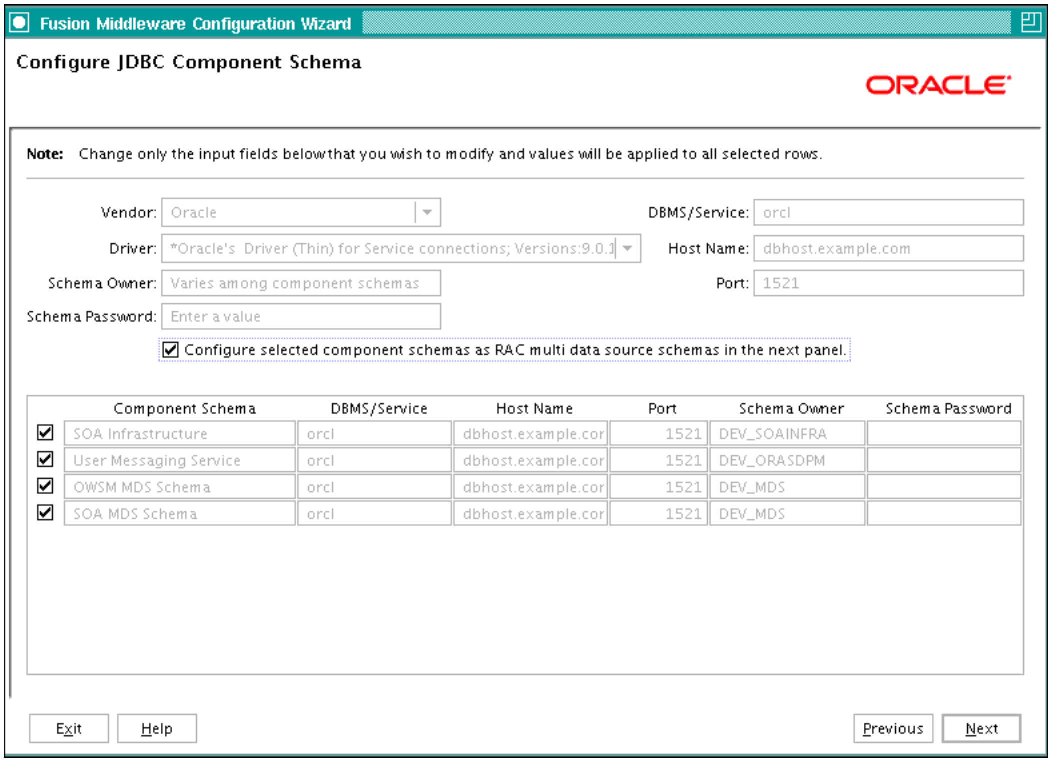
Click **Next**.

7. The Configure JDBC Component Schema screen opens. In this screen, do the following (as shown in Figure 6–2):

   - Select the following component schemas:
     - **SOA Infrastructure**
     - **User Messaging Service**
     - **OWSM MDS Schema**
     - **SOA MDS Schema**
   - Select **Configure selected component schemas as RAC multi data source schemas in the next panel**.

   Click **Next**.

Figure 6–2 Configure JDBC Component Schema Screen for Oracle SOA



Configure JDBC Component Schema

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

Vendor: Oracle

Driver: *Oracle's Driver (Thin) for Service connections; Versions:9.0.1

Schema Owner: Varies among component schemas

Schema Password: Enter a value

DBMS/Service: orcl

Host Name: dbhost.example.com

Port: 1521

☑ Configure selected component schemas as RAC multi data source schemas in the next panel.

	Component Schema	DBMS/Service	Host Name	Port	Schema Owner	Schema Password
☑	SOA Infrastructure	orcl	dbhost.example.cor	1521	DEV_SOAINFRA	
☑	User Messaging Service	orcl	dbhost.example.cor	1521	DEV_ORASDPM	
☑	OWSM MDS Schema	orcl	dbhost.example.cor	1521	DEV_MDS	
☑	SOA MDS Schema	orcl	dbhost.example.cor	1521	DEV_MDS	

8. The Configure RAC Multi Data Sources Component Schema screen opens. In this screen, enter values for the fields below, specifying the connect information for the Oracle RAC database that was seeded with RCU. Enter this information for each schema (you can select multiple schemas and specify values that are common to all):

> **Note:** Oracle recommends using the database used for identity management (see Chapter 11, "Integration with Oracle Identity Management") to store the Oracle WSM policies. It is therefore expected to use the IM database information has been seeded with RCU (as recommended in Chapter 2, "Database and Environment Preconfiguration") to store the WSM metadata and that this IM 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 SOA Suite schemas.)

- **Host name**, **instance name**, and **port** for the first Oracle RAC database instance. Then click **Add** and repeat for each Oracle RAC instance.
- **Driver:** Select **Oracle driver (Thin) for RAC Service-Instance connections, Versions: 10 and later**.
- **Service Name:** Enter the service name of the database (for example, **ecmedg.mycompany.com**).
- **Username:** Enter the complete user name (including the prefix) for the schemas. You can enter a value with all schemas selected (like the prefix) and then select each schema individually to change the rest of the schema name.
- **Password:** Enter the password to use to access the schemas.

Click **Next** when you are done.

9. In the Test JDBC Data Sources screen, the connections should be tested automatically. The Status column displays the results. Ensure that all connections were successful. If not, click **Previous** to return to the previous screen and correct your entries.

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

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

    - **JMS Distributed Destination**
    - **Managed Servers, Clusters and Machines**
    - **Deployment and Services**

    Click **Next**.

11. In the Select JMS Distributed Destination Type screen, accept the default destination type (UDD) for all resources (UMSJMSSystemResource and SOAJMSModule) and 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 called `soa_server1` is created automatically. Rename this to WLS_SOA1 and add a new server called WLS_SOA2. Give these server the attributes listed in Table 6–1. Do not modify the other servers that are shown in this screen; leave them as they are.

Table 6–1 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**.

13. In the Configure Clusters screen, add the clusters as listed in Table 6–2. Do not modify the other clusters that are shown in this screen; leave them as they are.

Table 6–2 Clusters

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

Click **Next**.

14. In the Assign Servers to Clusters screen, assign servers to clusters as shown below. Do not modify the other assignments that are shown in this screen; leave them as they are.

    - **SOA_Cluster:**
      - WLS_SOA1
      - WLS_SOA2

    Click **Next**.

15. 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 6–3 Machines

Name	Node Manager Listen Address
SOAHOST1	SOAHOST1
SOAHOST2	SOAHOST2
ADMINVHN	localhost

Leave all other fields to their default values. 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**.

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

    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 only to **SOA_Cluster**.
    - The **oracle.wsm.seedpolicies** library should be targeted only to **SOA_Cluster**.

    Click **Next**.

18. In the Target Services to Clusters or Servers screen, make sure that the **JOC Startup Class** and **JOC Shutdown Class** are targeted only to **SOA_Cluster**.

    Click **Next**.

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

    The domain is extended to include the SOA components.

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

21. Restart the Administration Server to enable these changes to take effect.

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

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

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

   ```
   SOAHOST1> cd ORACLE_COMMON_HOME/common/bin

   SOAHOST1> ./pack.sh -managed=true -domain=ORACLE_BASE/admin/domain_name/
   aserver/domain_name -template=edgdomaintemplate.jar -template_name=edgdomain_
   template
   ```

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

   ```
   SOAHOST1> cd ORACLE_COMMON_HOME/common/bin

   SOAHOST1> ./unpack.sh -domain=ORACLE_BASE/admin/domain_name/mserver/domain_name
   -template=edgdomaintemplate.jar -app_dir=ORACLE_BASE/admin/domain_name/mserver/
   applications
   ```

6.5 Configuring Oracle Coherence for Deploying Composites

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

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

Enabling Communication for Deployment Using Unicast Communication

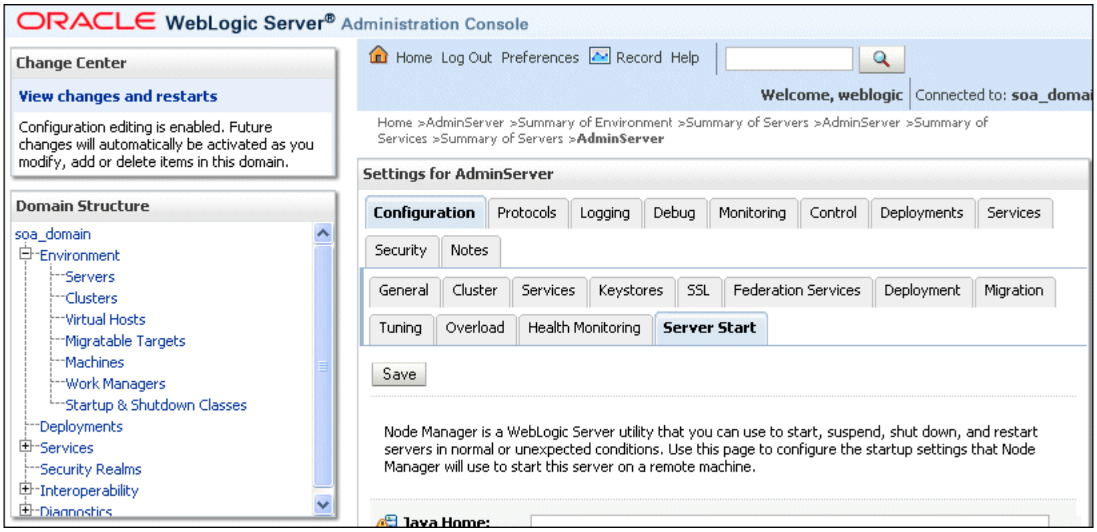
Multicast communication enables Oracle Fusion Middleware SOA to discover all of the members of a cluster to which it deploys composites dynamically. However, unicast communication does not enable nodes to discover other cluster members in this way. Consequently, you must specify the nodes that belong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

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

Specify the nodes using the `tangosol.coherence.wka`*`X`* system property, where *X* is a number between 1 and 9. You can specify up to nine nodes. Start the numbering at 1. This numbering must be sequential and must not contain gaps. In addition, specify the host name used by Oracle Coherence to create a cluster through the `tangosol.coherence.localhost` system property. This local host name should be the virtual host name used by the SOA server as the listener addresses (SOAHOST1VHN1 and SOAHOST2VHN1). Set this property by adding the `-Dtangosol.coherence.localhost` parameter to the Arguments field of the Oracle WebLogic Server Administration Console's Server Start tab (Figure 6–3).

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

Figure 6–3 Start Server Tab of Oracle WebLogic Server Administration Console



Configuration editing is enabled. Future changes will automatically be activated as you modify, add or delete items in this domain.

Node Manager is a WebLogic Server utility that you can use to start, suspend, shut down, and restart servers in normal or unexpected conditions. Use this page to configure the startup settings that Node Manager will use to start this server on a remote machine.

Specifying the host name

Perform these steps to add the host name used by Oracle Coherence:

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

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

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

Note: All arguments above are entered in one single line; that is, do not insert new lines in the Java arguments field in the Administration Console. Also, do not copy and paste the text from above to your Administration Console's arguments text field. This may result in extraneous characters being inserted in the Java arguments. The text should not contain other text or characters than the ones shown above.

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 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.wka1.port=8089
  -Dtangosol.coherence.wka2.port=8089
  -Dtangosol.coherence.localport=8089
  ```

- WLS_SOA2 (enter the following 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.wka1.port=8089
  -Dtangosol.coherence.wka2.port=8089
  -Dtangosol.coherence.localport=8089
  ```

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

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

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

6.6 Disabling Host Name Verification for the WLS_SOA Managed Servers

This step is required if you have not set up the appropriate certificates to authenticate the different nodes with the administration server (see Chapter 9, "Setting Up Node Manager"). If you have not configured the server certificates, you will receive errors when managing the different WebLogic Servers. To avoid these errors, disable host name verification while setting up and validating the topology, and enable it again once the EDG topology configuration is complete as described in Chapter 9, "Setting Up Node Manager."

Perform these steps to disable host name verification:

1. Log in to Oracle WebLogic Server Administration Console.
2. Click **Lock & Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page opens.
5. Select **WLS_SOA1** in the Names column of the table. The settings page for the server opens.
6. Open the SSL tab.
7. Expand the **Advanced** section of the page.
8. Set host name verification to 'None'.
9. Click **Save**.
10. Repeat steps 4 through 8 for the WLS_SOA2 managed server.
11. Save and activate the changes.
12. Restart Node Manager:
    1. Stop Node Manager by stopping the process associated with it.

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

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

       ```
       SOAHOST1> ps -ef | grep NodeManager
       orcl   9139  9120  0 Mar03 pts/6    00:00:00 /bin/sh ./startNodeManager.sh

       SOAHOST1>kill -9 9139
       ```

    2. Start Node Manager:

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

6.7 Starting and Validating the WLS_SOA1 Managed Server on SOAHOST1

Perform these steps to start the WLS_SOA1 managed server and check that it is configured correctly:

1. Start the WLS_SOA1 managed server using the Oracle WebLogic Server Administration Console as follows:
    1. Expand the Environment node in the Domain Structure window.
    2. Choose **Servers**. The Summary of Servers page opens.

   c. Click the Control tab.
   d. 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 12.8, "Troubleshooting" 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.

     Note: 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.

6.8 Propagating the Domain Configuration to SOAHOST2 Using the unpack Utility

Perform these steps to propagate the domain configuration:

1. Run the following command on SOAHOST1 to copy the template file created in Section 6.4, "Creating a Separate Domain Directory for Managed Servers in the Same Node as the Administration Server":

```
SOAHOST1> cd ORACLE_BASE/product/fmw/oracle_common/common/bin

SOAHOST1> scp edgdomaintemplate.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.

```
SOAHOST2> cd ORACLE_COMMON_HOME/common/bin

SOAHOST2> ./unpack.sh -domain=ORACLE_BASE/admin/domain_name/mserver/domain_name
-template=edgdomaintemplate.jar -app_dir=ORACLE_BASE/admin/domain_name/mserver/
applications
```

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

6.9 Starting Node Manager on SOAHOST2

Perform these steps to start Node Manager on SOAHOST2:

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

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

   > **Note:** You must use the `StartScriptEnabled` property to avoid class loading failures and other problems. See also Section 12.8.3, "Incomplete Policy Migration After Failed Restart of SOA Server."

2. Start Node Manager:

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

6.10 Starting and Validating the WLS_SOA2 Managed Server

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

1. Start the WLS_SOA2 managed server using the Oracle WebLogic Server Administration Console as follows:

   a. Expand the Environment node in the Domain Structure window.

   b. Choose **Servers**. The Summary of Servers page opens.

   c. Click 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 12.8, "Troubleshooting" 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.

6.11 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 (WSM). This local cache is provided to increase the performance of Oracle WSM. The Java Object Cache can be configured using the *MW_HOME*/oracle_common/bin/configure-joc.py script. This is a Python script which can be used to configure JOC in the managed servers. The script runs in WLST online mode and expects the Administration Server to be up and running.

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

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

Perform these steps 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 server URL, Oracle Weblogic Administration user name and password when prompted.

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

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

Specifically, for EDG environments, the first cluster option in step 2 should be used. Here is a walkthrough for using configure-joc.py for EDG environments (see below 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
```

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

Script Input Parameters

The input parameters are prompted by the script. The script can be used to perform the following 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'.

6.12 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. On WEBHOST1 and WEBHOST2, add the following lines to the *ORACLE_BASE*/admin/*instance_name*/config/OHS/*component_name*/mod_wl_ohs.conf file:

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

   # SOA soa-infra app
   <Location /soa-infra>
       WebLogicCluster SOAHOST1VHN1:8001,SOAHOST2VHN1:8001
       SetHandler weblogic-handler
   </Location>
   ```

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

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

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

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

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

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

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

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

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

> **Note:** Values such as ecm.mycompany.com:443, admin.mycompany.com:80 and you@youraddress that are noted in this document serve as examples only. Enter values based on the actual environment.

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

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

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

The servers specified in the `WebLogicCluster` parameter are only important at startup time for the plug-in. The list needs to provide at least one running cluster member for the plug-in to discover other members of the cluster. 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*.

6.13 Setting the Frontend HTTP Host and Port

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

1. Log in to Oracle WebLogic Server Administration Console.
2. Go to the Change Center section and click **Lock & Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Clusters**. The Summary of Clusters page opens.
5. Select **SOA_Cluster**.
6. Click 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 servers to make the frontend host directive in the cluster take effect.

Note: When HTTPS is enabled in the load balancer and the load balancer terminates SSL (the SOA servers receive only HTTP requests, not HTTPS), as suggested in this guide, the endpoint protocol for webservices is set to `http`. Since the load balancer redirects HTTP to HTTPS, this causes the following exception when testing web services functionality in Oracle Enterprise Manager Fusion Middleware Control:

```
(javax.xml.soap.SOAPException:
oracle.j2ee.ws.saaj.ContentTypeException)
```

To resolve this exception, update the URL endpoint:

In the Enterprise Manager Test Page, check **Edit Endpoint URL**.

Within the endpoint URL page:

- Change `http` to `https`.
- Change the default port number (say 80) to SSL port (say 443).

6.14 Validating Access Through Oracle HTTP Server

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 12.8, "Troubleshooting" for possible causes.

Validate WSM_Cluster through Oracle HTTP Server using the following URLs (for both WEBHOST1 and WEBHOST2):

- http://WEBHOST*n*:7777/soa-infra
- http://WEBHOST*n*:7777/integration/worklistapp
- http://WEBHOST*n*:7777/sdpmessaging/userprefs-ui
- http://WEBHOST*n*:7777/soa/composer
- http://WEBHOST*n*:7777/wsm-pm

For information on configuring system access through the load balancer, see Section 2.2.2, "Load Balancers."

Note: After the registering Oracle HTTP Server as described in Section 5.9, "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 Enterprise Manager Console. The WebTier item in the navigation tree should show that Oracle HTTP Server has been registered.

6.15 Configuring a Shared JMS Persistence Store

Configure the location for all of the persistence stores as a directory that is visible from both nodes. See Section 2.3, "Shared Storage and Recommended Directory Structure" for more information. You must change all of the persistent stores to use this shared base directory as follows:

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

   ORACLE_BASE/admin/*domain_name*/*soa_cluster_name*/jms

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

7. Click **Save and Activate**.
8. Restart the servers to make the change in the persistent stores take effect.

6.16 Configuring a Default Persistence Store for Transaction Recovery

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

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

Perform these steps to set the location for the default persistence store:

1. Log in to the Oracle WebLogic Server Administration Console.
2. In the Domain Structure window, expand the **Environment** node and then click the **Servers** node. The Summary of Servers page opens.
3. Click the name of the server (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.
4. Open the **Services** tab within the Configuration tab (not the top-level 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/soa_cluster_name/tlogs
   ```

7. Save and activate the changes.
8. Restart the WLS_SOA1 and WLS_SOA2 managed servers.

> **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.

6.17 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 by I/PM.

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

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

Using the Database Mutex Locking Operation

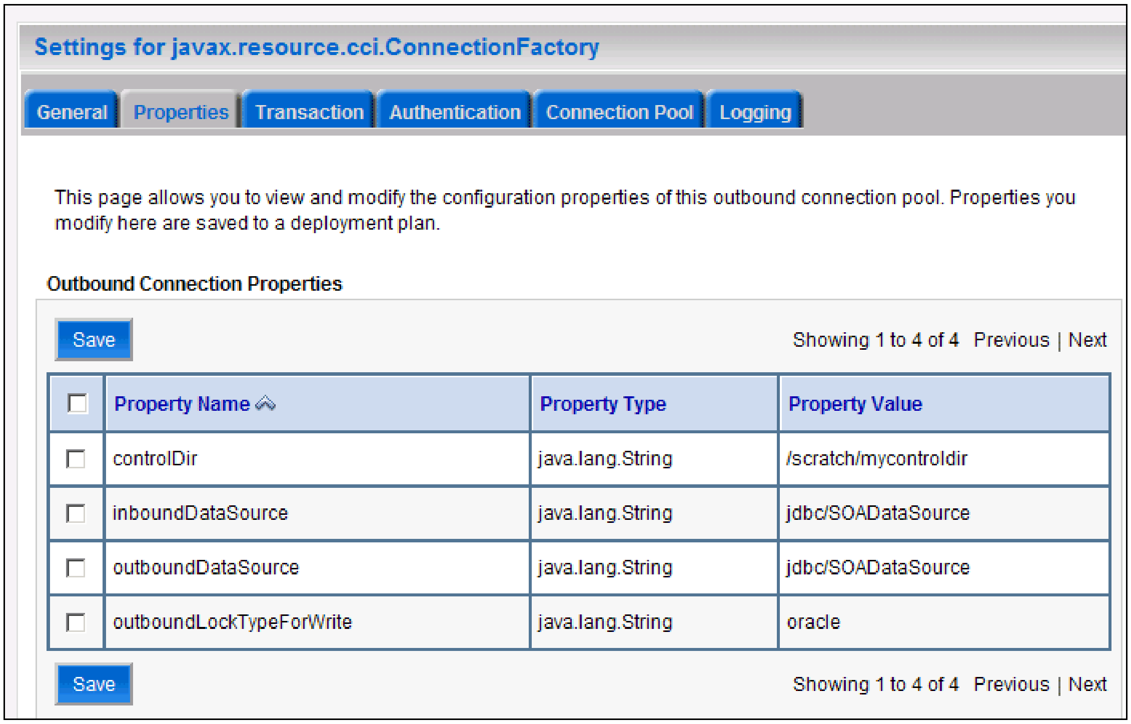
To make an outbound Oracle File or FTP Adapter service highly available using database table as a coordinator, you must modify the Oracle File Adapter deployment descriptor for the connection-instance corresponding to `eis/HAFileAdapter` in the Oracle WebLogic Server console:

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

1. Log in to the Oracle WebLogic Server console. To access the console, navigate to http://*server_name*:*port_number*/console.
2. Click **Deployments** in the left pane for Domain Structure.

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 6–4.

Figure 6–4 Oracle WebLogic Server Console - Settings for javax.resource.cci.Connectionfactory Page



Settings for javax.resource.cci.ConnectionFactory

This page allows you to view and modify the configuration properties of this outbound connection pool. Properties you modify here are saved to a deployment plan.

Outbound Connection Properties

□	Property Name	Property Type	Property Value
□	controlDir	java.lang.String	/scratch/mycontroldir
□	inboundDataSource	java.lang.String	jdbc/SOADataSource
□	outboundDataSource	java.lang.String	jdbc/SOADataSource
□	outboundLockTypeForWrite	java.lang.String	oracle

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

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

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

     ORACLE_BASE/admin/*domain_name*/*cluster_name*/fadapter

   - **inboundDataSource:** Set the value to 'jdbc/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. The Save Deployment Plan page opens.
9. 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. 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>
```

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

6.18 Scaling the Oracle Database Adapter

> **Note:** this step is optional and applies only to those deployments that require adapter support for the BPEL processes that are invoked by I/PM.

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 FMW 11*g*R1 PS1, 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`, 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 *Oracle Fusion Middleware User's Guide for Technology Adapters*.

6.19 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 9, "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

6.20 Configuring Server Migration for the WLS_SOA Managed Servers

Server migration is required for proper failover of the SOA components in the event of failure in any of the SOAHOST1 and SOAHOST2 nodes. See Chapter 10, "Configuring Server Migration" for further details. For SOA, 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

6.21 Backing Up the Installation

After you have verified that the extended domain is working, back up the installation. This is a quick backup for the express purpose of immediate restore in case of problems in the further steps. The backup destination is the local disk. This backup can be discarded once the enterprise deployment setup is complete. At 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. Also refer to the *Oracle Database Backup and Recovery Guide* for information on database backup.

Perform these steps to back up the installation at this point:

1. Back up the web tier:

   a. Shut down the instance using `opmnctl`:

   ```
   WEBHOST1> ORACLE_BASE/admin/instance_name/bin/opmnctl stopall
   ```

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

   ```
   WEBHOST1> tar -cvpf BACKUP_LOCATION/web.tar MW_HOME
   ```

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

   ```
   WEBHOST1> tar -cvpf BACKUP_LOCATION/web_instance_name.tar ORACLE_INSTANCE
   ```

   d. Start the instance using `opmnctl`:

   ```
   WEBHOST1> cd ORACLE_BASE/admin/instance_name/bin
   WEBHOST1> 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 to create the backup:

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

7 Extending the Domain with Oracle UCM

This chapter describes how to extend a domain with Oracle Universal Content Management (Oracle UCM) using the Oracle Fusion Middleware Configuration Wizard. It contains the following sections:

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

7.1 About Adding Oracle UCM to a Domain

The Oracle Enterprise Content Management Suite system is installed using the WL_HOME and ORACLE_HOME locations created in Chapter 3, "Installing the Software" on a shared storage. ECMHOST1 and ECMHOST2 mount MW_HOME and reuse the existing WLS, SOA, and ECM binary installations. The `pack` and `unpack` utilities are used to bootstrap the domain configuration for the WLS_UCM1 and WLS_UCM2 servers in these two new nodes. As a result, you do not need to install any software in these two nodes. For the Oracle ECM system to work properly, ECMHOST1 and ECMHOST2 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 SOA components to the domain as described in Section 6, "Extending the Domain with SOA Components."

7.2 Extending the Domain to Include Oracle UCM

You must extend the domain created in Section 5, "Creating a Domain with Administration Server" to include UCM. The instructions in this section assume that the ECM deployment uses the same database service as the SOA deployment (ecmedg.mycompany.com).

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

Perform these steps to extend the domain to include Oracle UCM:

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 check later on becomes more reliable.
2. Change the directory to the location of the Oracle Fusion Middleware Configuration Wizard. This is within the common home directory (notice that domain extensions are run from the node where the Administration Server resides).

   ```
   SOAHOST1> cd ORACLE_COMMON_HOME/common/bin
   ```

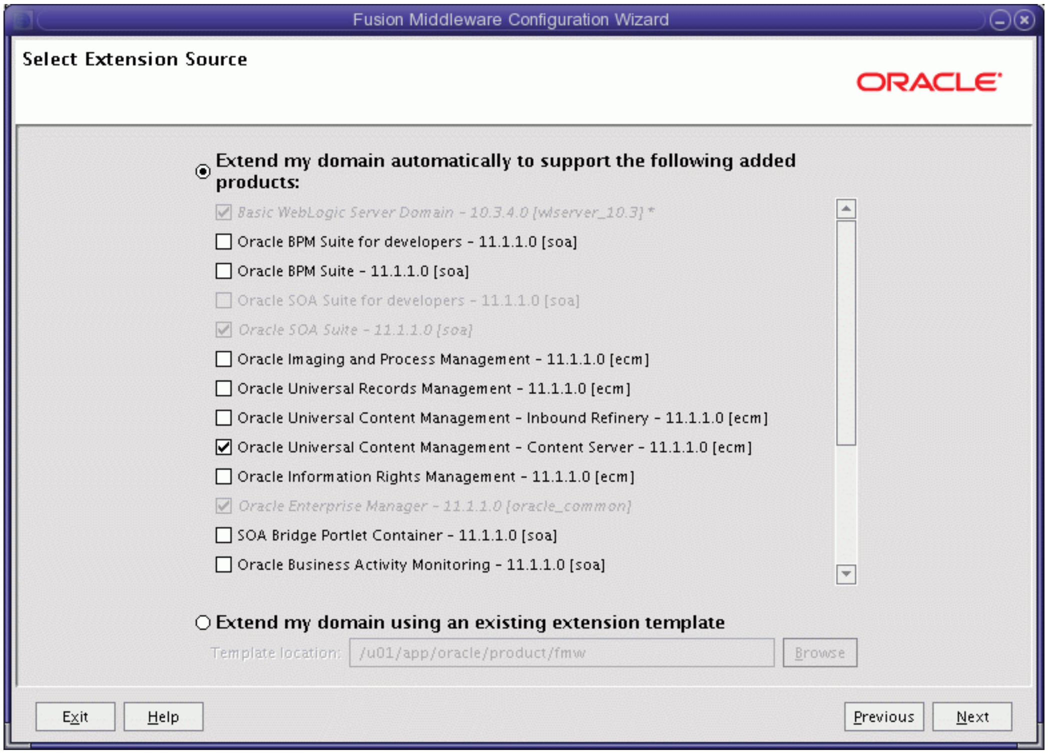
3. Start the Configuration Wizard:

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

4. In the Welcome screen, select **Extend an existing WebLogic domain**, and click **Next**.
5. In the WebLogic Domain Directory screen, select the WebLogic domain directory (*ORACLE_BASE*/admin/*domain_name*/aserver/*domain_name*), and click **Next**.
6. The Select Extension Source screen opens. In this screen, do the following (as shown in Figure 7–1):
   - Select **Extend my domain automatically to support the following added products**.
   - Select the following product:
     - **Oracle Universal Content Management - Content Server - 11.1.1.0 [ecm]**

   Click **Next**.

Figure 7–1 Select Extension Source screen for Oracle UCM



Fusion Middleware Configuration Wizard

Select Extension Source

● Extend my domain automatically to support the following added products:

- ☑ Basic WebLogic Server Domain - 10.3.4.0 [wlserver_10.3] *
- □ Oracle BPM Suite for developers - 11.1.1.0 [soa]
- □ Oracle BPM Suite - 11.1.1.0 [soa]
- □ Oracle SOA Suite for developers - 11.1.1.0 [soa]
- ☑ Oracle SOA Suite - 11.1.1.0 [soa]
- □ Oracle Imaging and Process Management - 11.1.1.0 [ecm]
- □ Oracle Universal Records Management - 11.1.1.0 [ecm]
- □ Oracle Universal Content Management - Inbound Refinery - 11.1.1.0 [ecm]
- ☑ Oracle Universal Content Management - Content Server - 11.1.1.0 [ecm]
- □ Oracle Information Rights Management - 11.1.1.0 [ecm]
- ☑ Oracle Enterprise Manager - 11.1.1.0 [oracle_common]
- □ SOA Bridge Portlet Container - 11.1.1.0 [soa]
- □ Oracle Business Activity Monitoring - 11.1.1.0 [soa]

○ Extend my domain using an existing extension template

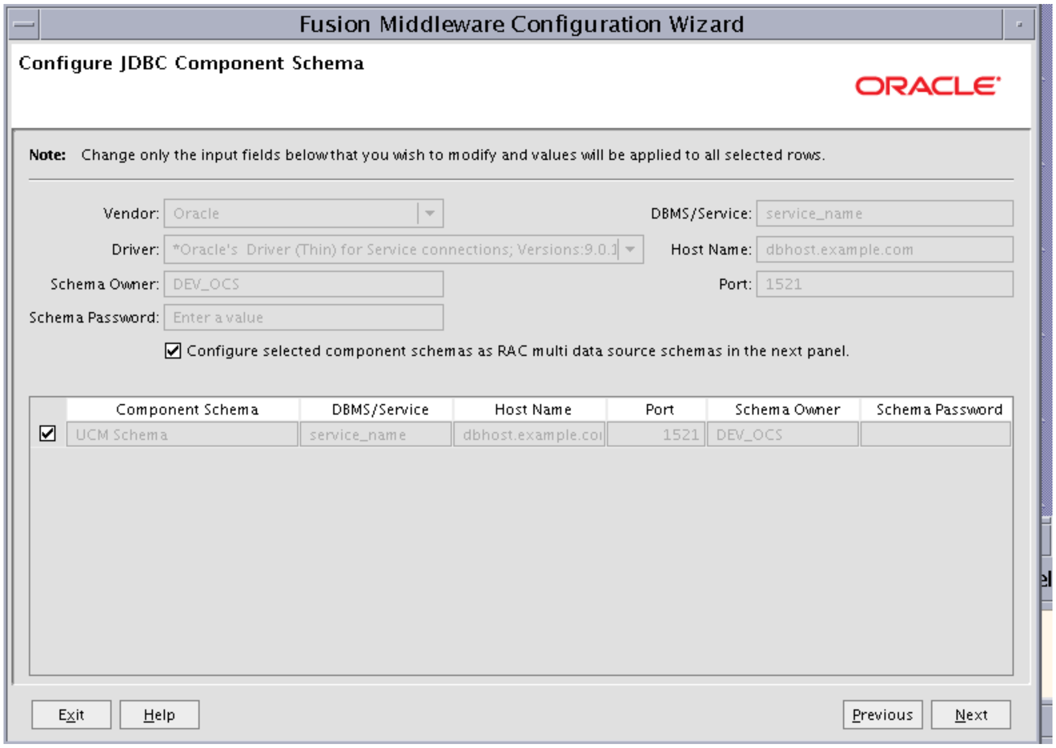
Template location: /u01/app/oracle/product/fmw

7. The Configure JDBC Component Schema screen opens. In this screen, do the following (as shown in Figure 7–2):

   - Select **UCM Schema**.
   - Select **Configure selected component schemas as RAC multi data source schemas in the next panel**.

   Click **Next**.

Figure 7–2 Configure JDBC Component Schema Screen for Oracle UCM



Fusion Middleware Configuration Wizard

Configure JDBC Component Schema

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

Vendor: Oracle

Driver: *Oracle's Driver (Thin) for Service connections; Versions:9.0.1

Schema Owner: DEV_OCS

Schema Password: Enter a value

DBMS/Service: service_name

Host Name: dbhost.example.com

Port: 1521

☑ Configure selected component schemas as RAC multi data source schemas in the next panel.

	Component Schema	DBMS/Service	Host Name	Port	Schema Owner	Schema Password
☑	UCM Schema	service_name	dbhost.example.coı	1521	DEV_OCS	

8. The Configure RAC Multi Data Sources Component Schema screen opens. In this screen, do the following (as shown in Figure 7–3):

   a. Select **UCM Schema**. Leave the other data sources as they are.

   b. 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 and later**.

      – **Service Name:** Enter the service name of the database (**ecmedg.mycompany.com**).

      – **Username:** Enter the complete user name (including the prefix) for the schemas. The user names shown in Figure 7–3 assume that `DEV` was used as the prefix for schema creation from RCU.

      – **Password:** Enter the password to use to access the schemas.

   c. Click **Add** and enter the details for the first Oracle RAC instance.

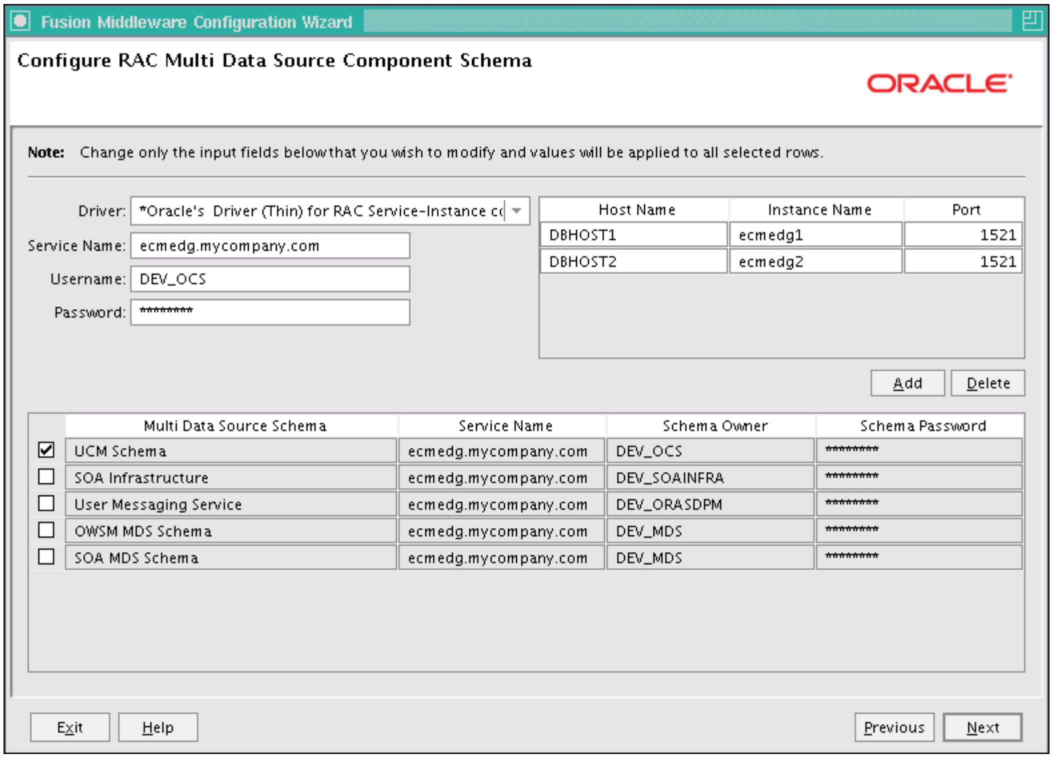
   d. Repeat step c for each Oracle RAC instance.

   Note: Leave the SOA Infrastructure, IPM Schema, User Messaging Service, OWSM MDS Schema, and SOA MDS Schema information as they are.

   ---

   e. Click **Next**.

Figure 7–3 Configure RAC Multi Data Source Component Schema Screen for Oracle UCM



Configure RAC Multi Data Source Component Schema

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

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

Service Name: ecmedg.mycompany.com

Username: DEV_OCS

Password: *********

Host Name	Instance Name	Port
DBHOST1	ecmedg1	1521
DBHOST2	ecmedg2	1521

	Multi Data Source Schema	Service Name	Schema Owner	Schema Password
☑	UCM Schema	ecmedg.mycompany.com	DEV_OCS	*********
☐	SOA Infrastructure	ecmedg.mycompany.com	DEV_SOAINFRA	*********
☐	User Messaging Service	ecmedg.mycompany.com	DEV_ORASDPM	*********
☐	OWSM MDS Schema	ecmedg.mycompany.com	DEV_MDS	*********
☐	SOA MDS Schema	ecmedg.mycompany.com	DEV_MDS	*********

9. In the Test JDBC Data Sources screen, the connections should be tested automatically. The **Status** column displays the results. Ensure that all connections were successful. If not, click **Previous** to return to the previous screen and correct your entries.

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

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

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

    Click **Next**.

11. In the Configure Managed Servers screen, click **Add** to add the required managed servers as shown in Table 7–1. Do not modify the other servers that appear in this screen; leave them as they are.

Table 7–1 Managed Servers

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled
WLS_UCM1	ECMHOST1	16200	n/a	No
WLS_UCM2	ECMHOST2	16200	n/a	No

Click **Next**.

12. In the Configure Clusters screen, click **Add** to add the clusters as shown in Table 7–2. Do not modify the other clusters that appear in this screen; leave them as they are.

Table 7–2 Clusters

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

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.

    - **UCM_Cluster:**
        - WLS_UCM1
        - WLS_UCM2

    Click **Next**.

14. In the Configure Machines screen, click the **Unix Machine** tab and add the following two new machines:

Table 7–3 Machines

Name	Node Manager Listen Address
ECMHOST1	ECMHOST1
ECMHOST2	ECMHOST2

Leave all other fields to their default values. Click **Next**.

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

    - Assign **WLS_UCM1** to **ECMHOST1**.
    - Assign **WLS_UCM2** to **ECMHOST2**.

    Click **Next**.

16. In the Target Deployments to Clusters or Servers screen, make sure that the DMS Application is targeted to the SOA_Cluster, UCM_Cluster and Admin Server. Click **Next**.

17. In the Target Services to Clusters or Servers screen, make sure the JOC Startup and JOC Shutdown classes are targeted to the SOA_Cluster only. 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. See Section 7.5, "Restarting the Administration Server."

7.3 Propagating the Domain Configuration to ECMHOST1 and ECMHOST2 Using the unpack Utility

Perform these steps to propagate the domain configuration:

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

   ```
   SOAHOST1> cd ORACLE_COMMON_HOME/common/bin

   SOAHOST1> ./pack.sh -managed=true -domain=ORACLE_BASE/admin/domain_name/
   aserver/domain_name -template=edgdomaintemplate2.jar -template_name=edgdomain_
   template
   ```

2. Copy the template to ECMHOST2:

   > **Note:** Assuming that ECMHOST1 shares the ORACLE_HOME with SOAHOST1, the template will be present in the same directory in ECMHOST1; otherwise, copy it also to ECMHOST1.

   ```
   SOAHOST1> scp edgdomaintemplate2.jar oracle@ECMHOST2:ORACLE_BASE/product/fmw/
   oracle_common/common/bin
   ```

3. Run the `unpack` command on ECMHOST1 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.

   ```
   ECMHOST1> cd ORACLE_COMMON_HOME/common/bin

   ECMHOST1> ./unpack.sh -domain=ORACLE_BASE/admin/domain_name/mserver/domain_name
   -template=edgdomaintemplate2.jar -app_dir=ORACLE_BASE/admin/domain_name/
   mserver/applications
   ```

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

4. Repeat step 3 for ECMHOST2.

7.4 Starting Node Manager on ECMHOST1 and ECMHOST2

Perform these steps to start Node Manager on ECMHOST1 and ECMHOST2 if Node Manager has not started already:

1. On each server, 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:

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

> **Note:** You must use the `StartScriptEnabled` property to avoid class loading failures and other problems. See also Section 12.8.3, "Incomplete Policy Migration After Failed Restart of SOA Server."

> **Note:** If the UCM server is sharing the MW_HOME in a local or shared storage with SOA, as suggested in the shared storage configuration described in Chapter 2, "Database and Environment Preconfiguration," 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 ECMHOST1 and ECMHOST2 to start Node Manager:

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

7.5 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 5.5, "Starting the Administration Server on SOAHOST1."

7.6 Starting and Configuring the WLS_UCM1 Managed Server

Starting the WLS_UCM1 Managed Server

Perform these steps to start the WLS_UCM1 managed server:

1. Start the WLS_UCM1 managed server using the Oracle WebLogic Server Administration Console as follows:

   a. Expand the Environment node in the Domain Structure window.

   b. Choose **Servers**. The Summary of Servers page opens.

   c. Click the Control tab.

   d. Select **WLS_UCM1** 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 12.8, "Troubleshooting" for possible causes.

Configuring the WLS_UCM1 Managed Server

Perform these steps to configure the WLS_UCM1 managed server:

1. Log in to WLS_UCM1 at http://ECMHOST1:16200/cs using your Oracle WebLogic administration user name and password to display a configuration page.

> **Note:** The UCM configuration files are on a shared disk so that all members of the cluster can access them. The shared disk location for the Oracle ECM enterprise deployment is at *ORACLE_BASE*/admin/ecm_domain/aserver/ucm_cluster.

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

   - **Content Server Instance Folder:** Set this to *ORACLE_BASE*/admin/ecm_domain/ucm_cluster/cs.
   - **Native File Repository Location:** Set this to *ORACLE_BASE*/admin/ecm_domain/ucm_cluster/cs/vault.
   - **WebLayout Folder:** Set this to *ORACLE_BASE*/admin/ecm_domain/ucm_cluster/cs/weblayout.
   - **Server Socket Port:** Set this to 4444.
   - **Socket Connection Address Security Filter:** Set this to a pipe-delimited list of localhost and the server IPs:

     ```
     127.0.0.1|ECMHOST1|ECMHOST2|WEBHOST1|WEBHOST2
     ```

     > **Note:** This will be changed later to a host name-based list (see Section 8.13, "Adding the Oracle I/PM Server Listen Addresses to the List of Allowed Hosts in Oracle UCM"). 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 'ecm.mycompany.com:443'.
   - **Web Address is HTTPS:** Select this check box.
   - **Server Instance Label:** Set this to 'UCM_Cluster1'.
   - **Server Instance Description:** Set this to 'Cluster ucm_cluster1'.
   - **Auto_Number Prefix:** Set this to 'ucm_cluster1-'.

3. Click **Submit** when finished and restart the managed server using the Oracle WebLogic Server Administration Console.

7.7 Updating the cwallet File in the Administration Server

The Oracle UCM 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 in SOAHOST1 using the following command (all on a single line):

```
ECMHOST1> 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 the WLS_UCM*n* servers 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.

7.8 Starting and Configuring the WLS_UCM2 Managed Server

Starting the WLS_UCM2 Managed Server

Perform these steps to start the WLS_UCM2 managed server:

1. Start the WLS_UCM2 managed server using the Oracle WebLogic Server Administration Console as follows:

   a. Expand the Environment node in the Domain Structure window.

   b. Choose **Servers**. The Summary of Servers page opens.

   c. Click the Control tab.

   d. Select **WLS_UCM2** 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 12.8, "Troubleshooting" for possible causes.

Configuring the WLS_UCM2 Managed Server

Perform these steps to configure the WLS_UCM2 managed server:

1. Log in to WLS_UCM2 at http://ECMHOST2:16200/cs using your Oracle WebLogic administration user name and password to display a configuration page.

   Note: The UCM configuration files are on a shared disk so that all members of the cluster can access them. The shared disk location for the Oracle ECM enterprise deployment is at *ORACLE_BASE*/admin/ecm_domain/aserver/ucm_cluster.

2. Change the following values on the server configuration page:

   - **Content Server Instance Folder:** Set this to *ORACLE_BASE*/admin/ecm_domain/ucm_cluster/cs

   - **Native File Repository Location:** Set this to *ORACLE_BASE*/admin/ecm_domain/ucm_cluster/cs/vault

   - **WebLayout Folder:** Set this to *ORACLE_BASE*/admin/ecm_domain/ucm_cluster/cs/weblayout

   Make sure that the 'Is new Content Server Instance?' check box is not selected.

3. Click **Submit** when finished and restart the managed server using the Oracle WebLogic Server Administration Console.

7.9 Configuring Service Retries for Oracle UCM

The following parameter should be set in Oracle Content Server's config.cfg file in order 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.

Perform these steps to add the configuration parameter for Oracle UCM:

1. Go to the WebLogic Server Administration Console for Oracle UCM at http://ECMHOST1:16200/cs, and log in using your Oracle WebLogic administration user name and password.
2. Open the Administration page, and then choose **Admin Server**. The Content Admin Server page opens.
3. Click **General Configuration** on the left. The General Configuration page opens.
4. In the Additional Configuration Variables box, add the parameter:

   ```
   ServiceAllowRetry=true
   ```

5. Click **Save** and restart all UCM managed servers.

> **Note:** The new parameter is included in the config.cfg file, which is at the following location:
>
> ```
> ORACLE_BASE/admin/ecm_domain/ucm_cluster/cs/config/config.cfg
> ```
>
> (You can also edit this file directly in a text editor. Do not forget to restart all UCM managed servers.)

7.10 Configuring Oracle HTTP Server for the WLS_UCM Managed Servers

To enable Oracle HTTP Server to route to UCM_Cluster, which contain the WLS_UCM1 and WLS_UCM2 managed servers, you must set the `WebLogicCluster` parameter to the list of nodes in the cluster:

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

   ```
   # UCM
   <Location /cs>
      WebLogicCluster ECMHOST1:16200,ECMHOST2:16200
      SetHandler weblogic-handler
      WLCookieName JSESSIONID
      WLProxySSL ON
      WLProxySSLPassThrough ON
   </Location>

   <Location /adfAuthentication>
      WebLogicCluster ECMHOST1:16200,ECMHOST2:16200
      SetHandler weblogic-handler
      WLCookieName JSESSIONID
      WLProxySSL ON
      WLProxySSLPassThrough ON
   </Location>
   ```

```
<Location /_ocsh>
   WebLogicCluster ECMHOST1:16200,ECMHOST2:16200
   SetHandler weblogic-handler
   WLCookieName JSESSIONID
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>
```

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

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

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

7.11 Validating Access Through Oracle HTTP Server

You should verify URLs to ensure that appropriate routing and failover is working from Oracle HTTP Server to UCM_Cluster. Perform these steps to verify the URLs:

1. While WLS_UCM2 is running, stop WLS_UCM1 using the Oracle WebLogic Server Administration Console.
2. Access http://WEBHOST1:7777/cs to verify it is functioning properly.
3. Start WLS_UCM1 from the Oracle WebLogic Server Administration Console.
4. Stop WLS_UCM2 from the Oracle WebLogic Server Administration Console.
5. Access http://WEBHOST1:7777/cs to verify it is functioning properly.

7.12 Configuring Node Manager for the WLS_UCM and WLS_IPM 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 9, "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:	ECMHOST1	ECMHOST1VHN1*	WLS_UCM1
Run 2:	ECMHOST2	ECMHOST2VHN1*	WLS_UCM2

* Optional; required only for WLS-managed Oracle I/PM servers (WLS_IPM).

Please note the following:

- Even though the WLS_IPM managed servers are not yet configured at this point (see Chapter 8, "Extending the Domain with Oracle I/PM") and are not mandatory for a UCM-only Oracle ECM installation, the virtual host names used by Oracle I/PM are configured here in order to provide a one-step configuration process that includes both types of servers.

- Perform all steps in Chapter 9, "Setting Up Node Manager" except for Section 9.3.5, "Configuring Managed WLS Servers to Use the Custom Keystores" for the WLS_IPM servers. This step can be done after the WLS_IPM servers are added to the domain.

7.13 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. Also refer to the *Oracle Database Backup and Recovery Guide* for information on database backup.

Perform these steps to back up the installation at this point:

1. Back up the web tier:

   a. Shut down the instance using `opmnctl`.

   ```
   WEBHOST1> ORACLE_BASE/admin/instance_name/bin/opmnctl stopall
   ```

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

   ```
   WEBHOST1> tar -cvpf BACKUP_LOCATION/web.tar MW_HOME
   ```

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

   ```
   WEBHOST1> tar -cvpf BACKUP_LOCATION/web_instance_name.tar ORACLE_INSTANCE
   ```

   d. Start the instance using `opmnctl`:

   ```
   WEBHOST1> cd ORACLE_BASE/admin/instance_name/bin
   WEBHOST1> 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 to create the backup:

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

8

Extending the Domain with Oracle I/PM

This chapter describes how to extend a domain with Oracle Imaging and Processing Management (Oracle I/PM) using the Oracle Fusion Middleware Configuration Wizard. It contains the following sections:

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

8.1 About Adding Oracle I/PM to a Domain

The Oracle Imaging and Processing Management (I/PM) system is installed using the WL_HOME and ORACLE_HOME locations created in Chapter 3, "Installing the Software" on a shared storage. ECMHOST1 and ECMHOST2 mount MW_HOME and reuse the existing WLS, SOA, and ECM binary installations. The `pack` and `unpack` utilities are used to bootstrap the domain configuration for the WLS_IPM1 and WLS_IPM2 servers in these two new nodes. As a result, you do not need to install any software in these two nodes. For the I/PM system to work properly, ECMHOST1 and ECMHOST2 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.

8.2 Enabling VIP4 and VIP5 in ECMHOST1and ECMHOST2

The I/PM system uses a virtual host name as the listen addresses for the managed server on which it is running. These virtual host names and corresponding virtual IPs are required to enable server migration for the I/PM component. You must enable a VIP (VIP4/VIP5) mapping to ECMHOST1VHN1 on ECMHOST1 and ECMHOST2VHN1 on ECMHOST2, and must also correctly resolve the host names in the network system used by the topology (either by DNS Server or hosts resolution).

To enable the VIPs, follow the example described in Section 6.2, "Enabling SOAHOST1VHN1 on SOAHOST1 and SOAHOST2VHN1 on SOAHOST2."

8.3 Extending the Domain to Include Oracle I/PM

You extend the domain configured in Chapter 7, "Extending the Domain with Oracle UCM" to include Oracle I/PM. The instructions in this section assume that the Oracle I/PM deployment uses the same database service as the Oracle UCM deployment (ecmedg.mycompany.com). However, a deployment may choose to use a different database service specifically for Oracle I/PM.

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

Perform these steps to extend the domain to include Oracle I/PM:

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 check later on becomes more reliable.

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

   ```
   SOAHOST1> cd ORACLE_COMMON_HOME/common/bin
   ```

3. Start the Configuration Wizard:

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

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

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

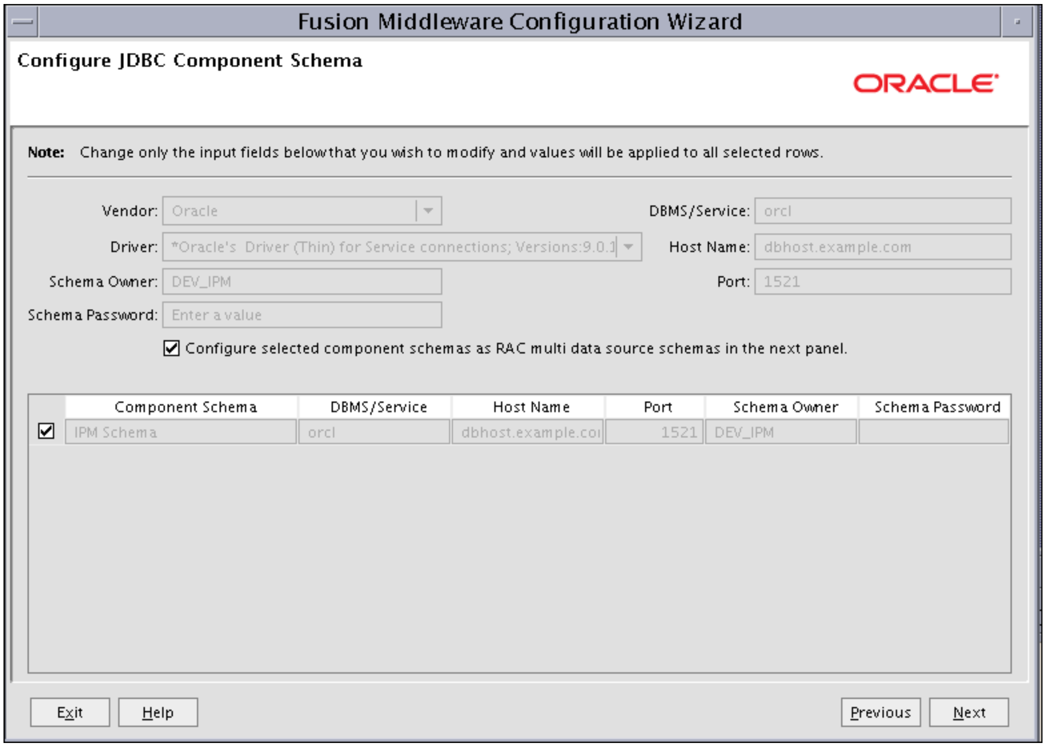
6. In the Select Extension Source screen, do the following:

   - Select **Extend my domain automatically to support the following added products**.
   - Select the following product:
     – **Oracle Image and Process Management - 11.1.1.0 [ecm]**

   Click **Next**.

7. The Configure JDBC Component Schema screen opens (Figure 8–1).

Figure 8–1 Configure JDBC Component Schema Screen for Oracle I/PM



Fusion Middleware Configuration Wizard

Configure JDBC Component Schema

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

Vendor: Oracle
Driver: *Oracle's Driver (Thin) for Service connections; Versions:9.0.1
Schema Owner: DEV_IPM
Schema Password: Enter a value
DBMS/Service: orcl
Host Name: dbhost.example.com
Port: 1521

☑ Configure selected component schemas as RAC multi data source schemas in the next panel.

	Component Schema	DBMS/Service	Host Name	Port	Schema Owner	Schema Password
☑	IPM Schema	orcl	dbhost.example.coi	1521	DEV_IPM	

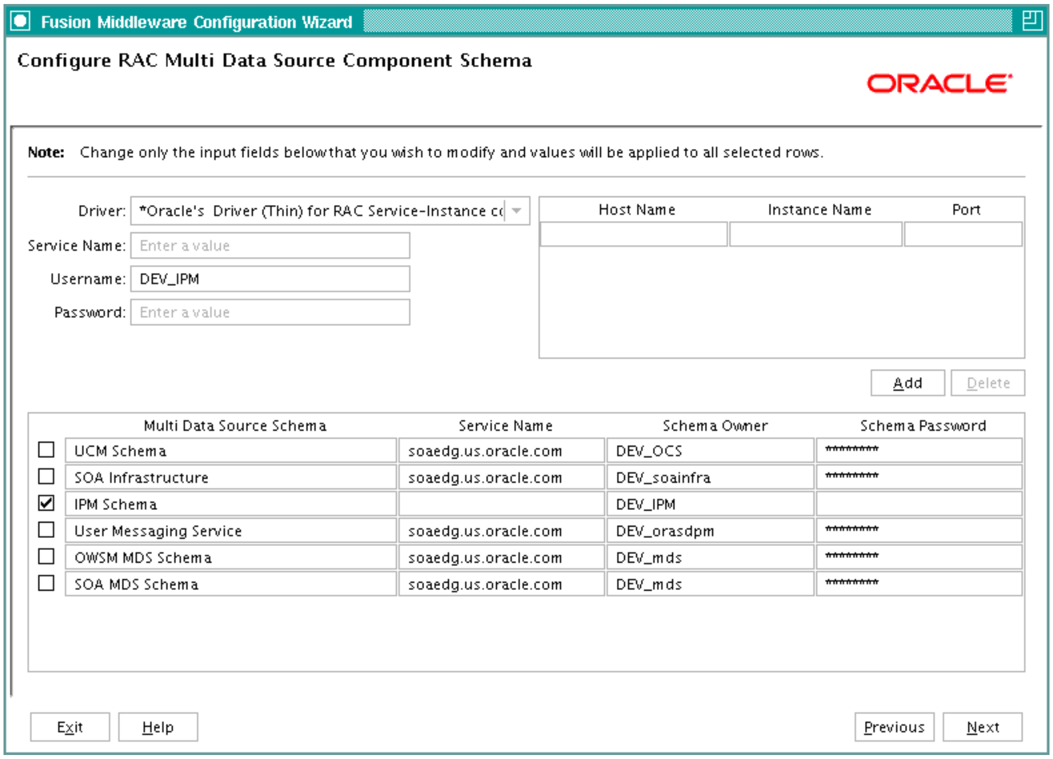
In the Configure JDBC Component Schema screen, do the following:

- Select **IPM Schema** only. Do not select any of the other existing schemas!
- Select **Configure selected component schemas as RAC multi data source schemas in the next panel**.

Click **Next**.

8. The Configure RAC Multi Data Sources Component Schema screen opens (Figure 8–2).

Figure 8–2 Configure RAC Multi Data Source Component Scheme Screen for Oracle I/PM



Configure RAC Multi Data Source Component Schema

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

Driver: *Oracle's Driver (Thin) for RAC Service-Instance co

Service Name: Enter a value

Username: DEV_IPM

Password: Enter a value

Host Name	Instance Name	Port

	Multi Data Source Schema	Service Name	Schema Owner	Schema Password
□	UCM Schema	soaedg.us.oracle.com	DEV_OCS	**********
□	SOA Infrastructure	soaedg.us.oracle.com	DEV_soainfra	**********
☑	IPM Schema		DEV_IPM	
□	User Messaging Service	soaedg.us.oracle.com	DEV_orasdpm	**********
□	OWSM MDS Schema	soaedg.us.oracle.com	DEV_mds	**********
□	SOA MDS Schema	soaedg.us.oracle.com	DEV_mds	**********

In the Configure RAC Multi Data Sources Component Schema screen, do the following:

a. Select **IPM Schema**. Leave the other data sources as they are.

b. 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 and later**.
- **Service Name:** Enter the service name of the database (**ecmedg.mycompany.com**).
- **Username:** Enter the complete user name (including the prefix) for the schemas. The user names shown in Figure 8–2 assume that `DEV` was used as the prefix for schema creation from RCU.
- **Password:** Enter the password to use to access the schemas.

c. Click **Add** and enter the details for the first Oracle RAC instance.

d. Repeat step c for each Oracle RAC instance.

> **Note:** Leave the UCM Schema, SOA Infrastructure, User Messaging Service, OWSM MDS Schema, and SOA MDS Schema information as they are.

e. Click **Next**.

9. In the Test JDBC Data Sources screen, the connections should be tested automatically. The **Status** column displays the results. Ensure that all connections were successful. If not, click **Previous** to return to the previous screen and correct your entries.

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

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

    - **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 called 'ipm_server1' is created automatically. Rename this to WLS_IPM1 and add a new server called WLS_IPM2. Give these servers the attributes listed in Table 8–1. Do not modify the other servers that appear in this screen; leave them as they are.

Table 8–1 Managed Servers

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled
WLS_IPM1	ECMHOST1VHN1	16000	n/a	No
WLS_IPM2	ECMHOST2VHN2	16000	n/a	No

    Click **Next**.

13. In the Configure Clusters screen, click **Add** to add the clusters as shown in Table 8–2. Do not modify the other clusters that appear in this screen; leave them as they are.

Table 8–2 Clusters

Name	Cluster Messaging Mode	Multicast Address	Multicast Port	Cluster Address
IPM_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.

    - **IPM_Cluster:**
      - WLS_IPM1
      - WLS_IPM2

    Click **Next**.

15. In the Configure Machines screen, click the **Unix Machine** tab. You should see the ECMHOST1 and ECMHOST2 machines and have the following entries:

Table 8–3 Machines

Name	Node Manager Listen Address
SOAHOST1	SOAHOST1
SOAHOST2	SOAHOST2
ADMINVHN	localhost
ECMHOST1	ECMHOST1
ECMHOST2	ECMHOST2

    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_IPM1** to **ECMHOST1**.
    - Assign **WLS_IPM2** to **ECMHOST2**.

    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 **IPM_Cluster**.
    - The **oracle.wsm.seedpolicies** library should be targeted to **SOA_Cluster** and **IPM_Cluster** (and any servers expected to host WSM-PM protected web services).

    Click **Next**.

18. In the Target Service to Cluster or Servers screen, target **JOC Startup Class** and **JOC Shutdown Class** only to **SOA_Cluster**.

    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. See Section 8.6, "Restarting the Administration Server."

8.4 Propagating the Domain Configuration to ECMHOST1 and ECMHOST2 Using the unpack Utility

Perform these steps to propagate the domain configuration:

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

```
SOAHOST1> cd ORACLE_COMMON_HOME/common/bin

SOAHOST1> ./pack.sh -managed=true -domain=ORACLE_BASE/admin/domain_name/
aserver/domain_name -template=edgdomaintemplateIPM.jar -template_
name=edgdomain_templateIPM
```

2. Copy the template to ECMHOST2:

> **Note:** Assuming that ECMHOST1 shares the ORACLE_HOME with SOAHOST1, the template will be present in the same directory in ECMHOST1; otherwise, copy it also to ECMHOST1.

```
SOAHOST1> scp edgdomaintemplateIPM.jar oracle@ECMHOST2:ORACLE_BASE/product/fmw/
oracle_common/common/bin
```

3. Run the `unpack` command on ECMHOST1 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.

```
ECMHOST1> cd ORACLE_COMMON_HOME/common/bin

ECMHOST1> ./unpack.sh -domain=ORACLE_BASE/admin/domain_name/mserver/domain_name
-template=edgdomaintemplateIPM.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 running `unpack`. In addition, the *ORACLE_BASE*/admin/*domain_name*/mserver/applications must be empty.

4. Repeat step 3 for ECMHOST2.

8.5 Starting Node Manager on ECMHOST1 and ECMHOST2

Perform these steps to start Node Manager on ECMHOST1 and ECMHOST2 if Node Manager has not started already:

1. On each server, 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:

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

> **Note:** You must use the `StartScriptEnabled` property to avoid class loading failures and other problems. See also Section 12.8.3, "Incomplete Policy Migration After Failed Restart of SOA Server."

> **Note:** If the I/PM server is sharing the MW_HOME in a local or shared storage with UCM, as suggested in the shared storage configuration described in Chapter 2, "Database and Environment Preconfiguration," it is not required to run setNMProps.sh again. In this case, Node Manager has already been configured to use a start script and it is likely already running in the node for UCM.

2. Run the following commands on both ECMHOST1 and ECMHOST2 to start Node Manager:

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

8.6 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 5.5, "Starting the Administration Server on SOAHOST1."

8.7 Configuring a JMS Persistence Store for Oracle I/PM 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 I/PM are configured with no persistent store and use WebLogic Server's store (*ORACLE_BASE*/admin/*domain_name*/mserver/*domain_name*/servers/*server_name*/data/store/default). You must change I/PM's JMS server persistent store to use a shared base directory as follows:

1. Log in to the Oracle WebLogic Server Administration Console.
2. In the Domain Structure window, expand the **Services** node and then click the **Persistence Stores** node. The Summary of Persistence Stores page opens.
3. Click **Lock & Edit**.
4. Click **New**, and then **Create File Store**.
5. Enter a **name** (for example 'IPMJMSServer1Store', which allows you identify the service it is created for) and target WLS_IPM1. Enter a **directory** that is located in shared storage so that it is accessible from both ECMHOST1 and ECMHOST2 (*ORACLE_BASE*/admin/*domain_name*/ipm_cluster/jms).
6. Click **OK** and activate the changes.
7. In the Domain Structure window, expand the **Services** node and then click the **Messaging->JMS Servers** node. The Summary of JMS Servers page opens.
8. Click on the IpmJmsServer1 JMS Server (represented as a hyperlink) from the **Name** column of the table. The Settings page for the JMS server opens.
9. Click **Lock & Edit**.
10. In the Persistent Store drop-down list, select **IPMJMSServer1Store**.

11. Click **Save and Activate**.
12. Repeat the steps and create **IPMJMSServer2Store** for IPMJMSServer2.

8.8 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. The WebLogic Server uses this transaction log for recovery from system crashes or network failures. To leverage the migration capability of the Transaction Recovery Service for the servers within a cluster, store the transaction log in a location accessible to the server.

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

Perform these steps to set the location for the default persistence store for WLS_IPM1:

1. Log in to the Oracle WebLogic Server Administration Console.
2. In the Domain Structure window, expand the **Environment** node and then click the **Servers** node. The Summary of Servers page opens.
3. Click WLS_IPM1 (represented as a hyperlink) in the **Name** column of the table. The settings page for the WLS_IPM1 server opens with the Configuration tab active.
4. Click 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/ipm_cluster_name/tlogs
   ```

7. Click **Save** and activate the changes.
8. Repeat the step for the WLS_IPM2 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 ECMHOST1 and ECMHOST2 must be able to access this directory. This directory must also exist before you restart the server.

8.9 Disabling Host Name Verification for the WLS_IPM Managed Servers

This step is required if you have not set up the appropriate certificates to authenticate the different nodes with the Administration Server (see Chapter 9, "Setting Up Node Manager"). If you have not configured the server certificates, you will receive errors when managing the different WebLogic Servers. To avoid these errors, disable host name verification while setting up and validating the topology, and enable it again once the EDG topology configuration is complete as described in Chapter 9, "Setting Up Node Manager."

Perform these steps to disable host name verification:

1. Log in to Oracle WebLogic Server Administration Console.
2. In the Administration Console, select **WLS_IPM1**, then **SSL**, and then **Advanced**.
3. Click **Lock & Edit**.
4. Set host name verification to 'None'.
5. In the Administration Console, select **WLS_IPM2**, then **SSL**, and then **Advanced**.
6. Set host name verification to 'None'.
7. Save and activate the changes.

8.10 Starting the Oracle I/PM System

Perform these steps to start the WLS_IPM1 managed server on ECMHOST1:

1. Start the WLS_IPM1 managed servers:
   a. Log in to the Oracle WebLogic Server Administration Console at http://ADMINVHN:7001/console.
   b. In the Domain Structure window, expand the **Environment** node and then select **Servers**. The Summary of Servers page opens.
   c. Click the **Control** tab.
   d. Select WLS_IPM1 from the **Servers** column of the table.
   e. Click **Start**.
2. Access http://ECMHOST1VHN1:16000/imaging to verify the status of WLS_IPM1. The Imaging and Process Management login page appears. Enter your Oracle Weblogic administration user name and password to log in.

   Verify that the PROCESSES initialization parameter for the database is set to a high enough value. See Section 2.1.1.3, "Initialization Parameters" for details. This error often occurs when you start servers that are subsequent to the first server.
3. Start the WLS_IPM2 managed servers:
   a. Log in to the Oracle WebLogic Server Administration Console at http://ADMINVHN:7001/console.
   b. In the Domain Structure window, expand the **Environment** node and then select **Servers**. The Summary of Servers page opens.
   c. Click the **Control** tab.
   d. Select WLS_IPM2 from the **Servers** column of the table.
   e. Click **Start**.
4. Access http://ECMHOST2VHN1:16000/imaging to verify the status of WLS_IPM2. The Imaging and Process Management login page appears. Enter your Oracle Weblogic administration user name and password to log in.

> **Note:** These instructions assume that the host name verification displayed previously for the WS-M or SOA managed servers in SOAHOST2 and that the Node Manager is already running on SOAHOST2.

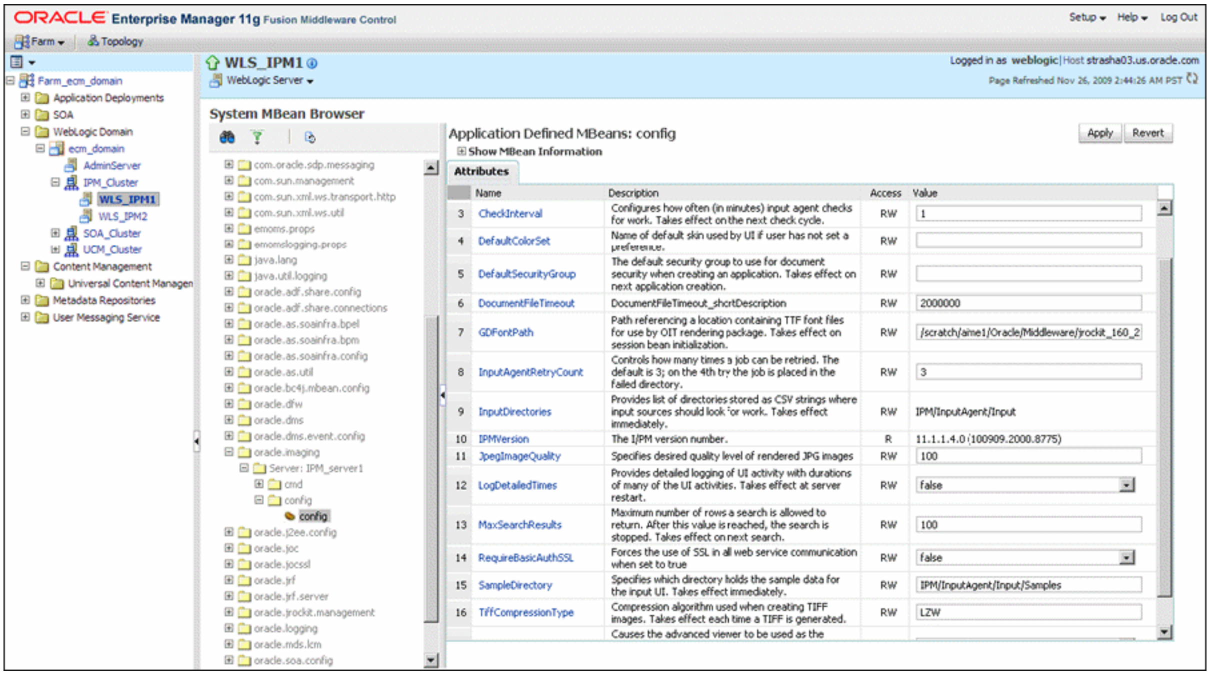
8.11 Configuring System MBeans for Oracle I/PM

Perform these steps to configure the following system MBeans for Oracle I/PM:

- InputDirectories
- SampleDirectory
- GDFontPath

1. Log in to Oracle Fusion Middleware Control at http://ADMINVHN:7001/em (Figure 8–3).

Figure 8–3 System MBean Browser



	Name	Description	Access	Value
3	CheckInterval	Configures how often (in minutes) input agent checks for work. Takes effect on the next check cycle.	RW	1
4	DefaultColorSet	Name of default skin used by UI if user has not set a preference.	RW	
5	DefaultSecurityGroup	The default security group to use for document security when creating an application. Takes effect on next application creation.	RW	
6	DocumentFileTimeout	DocumentFileTimeout_shortDescription	RW	2000000
7	GDFontPath	Path referencing a location containing TTF font files for use by OIT rendering package. Takes effect on session bean initialization.	RW	/scratch/aime1/Oracle/Middleware/jrockit_160_2
8	InputAgentRetryCount	Controls how many times a job can be retried. The default is 3; on the 4th try the job is placed in the failed directory.	RW	3
9	InputDirectories	Provides list of directories stored as CSV strings where input sources should look for work. Takes effect immediately.	RW	IPM/InputAgent/Input
10	IPMVersion	The I/PM version number.	R	11.1.1.4.0 (100909.2000.8775)
11	JpegImageQuality	Specifies desired quality level of rendered JPG images	RW	100
12	LogDetailedTimes	Provides detailed logging of UI activity with durations of many of the UI activities. Takes effect at server restart.	RW	false
13	MaxSearchResults	Maximum number of rows a search is allowed to return. After this value is reached, the search is stopped. Takes effect on next search.	RW	100
14	RequireBasicAuthSSL	Forces the use of SSL in all web service communication when set to true	RW	false
15	SampleDirectory	Specifies which directory holds the sample data for the input UI. Takes effect immediately.	RW	IPM/InputAgent/Input/Samples
16	TiffCompressionType	Compression algorithm used when creating TIFF images. Takes effect each time a TIFF is generated.	RW	LZW
		Causes the advanced viewer to be used as the		

2. In the left pane, expand the farm domain name, then expand **WebLogic Domain**, then the domain name, then **IPM Cluster**, and then click the "WLS_IPM1" link.

3. At the top of the right-hand panes, click on the WebLogic Server drop-down menu and choose System MBean Browser.

4. Expand **Application Defined MBeans** and then **oracle.imaging**.

5. Expand **Server: WLS_IPM1** 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*/*ipm_cluster_name*/input_files.

   Please note that all Oracle UCM servers involved must be able to resolve this location (that is, via the NFS mount point).

8. Set the **SampleDirectory** MBean: *ORACLE_BASE*/admin/*domain_name*/ *ipm_cluster_name*/input_files/Samples.

   In order to process input files, the input agent must have the appropriate permissions on 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 to the input directory and all files and subdirectories in the input directory. These privileges are required so that input agent can move the files to the various directories as it works on them. File locking on the share is needed by 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 defaults are likely "/usr/share/X11/fonts/TTF" or "/usr/lib/X11/fonts/TTF ."
10. Click **Apply**.

8.12 Enabling Oracle I/PM in Oracle UCM

Perform these steps to enable Oracle I/PM in Oracle UCM:

1. Log in to Oracle Content Server at http://ECMHOST1:16200/cs.
2. Open the **Administration** tray or menu, and choose **Admin Server**.
   The Component Manager page opens.
3. Enable the IpmRepository component.
4. Click **Update** and confirm the action.
5. Restart the managed server, and then restart all managed servers in the UCM cluster.

8.13 Adding the Oracle I/PM Server Listen Addresses to the List of Allowed Hosts in Oracle UCM

Perform these steps to add the host names of the WLS_IPM1 and WLS_IPM2 managed servers (ECMHOST1VHN1 and ECMHOST2VHN1, respectively) to the SocketHostNameSecurityFilter parameter list:

1. Open the file *ORACLE_BASE*/admin/*domain_name*/ucm_cluster/cs/config/config.cfg in a text editor.
2. Remove or comment out the following line:

```
SocketHostAddressSecurityFilter=127.0.0.1|ECMHOST1|ECMHOST2|WEBHOST1|WEBHOST2
```

3. Add the following two lines to include the WLS_IPM1 and WLS_IPM2 listen addresses to the list of addresses that are allowed to connect to Oracle UCM:

```
SocketHostNameSecurityFilter=localhost|localhost.mycompany.com|ECMHOST1|
ECMHOST2|ECMHOST1VHN1|ECMHOST2VHN1
AlwaysReverseLookupForHost=Yes
```

4. Save the modified config.cfg file and restart the UCM servers for the changes to take effect.

8.14 Creating a Connection to UCM System

Perform these steps to create a connection to the Oracle UCM system:

1. Log in to WLS_IPM1 imaging console at http://ECMHOST1VHN1:16000/imaging.
2. In the left-hand pane, 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** check box is selected.
   - Set the Content Server port to 4444.
   - Add two servers to the Content Server pool:
     - ECMHOST1:4444
     - ECMHOST2:4444

   Click **Next**.
5. In the Connection Security screen, leave the default selections for the WebLogic user, and then click **Next**.
6. Review the connection details and click **Submit**.

8.15 Configuring BPEL CSF Credentials

When connecting to a BPEL system from Oracle I/PM, it is required to configure the required credential to communicate with the SOA system. To add these credentials, use these steps:

1. Change directory to the common/bin location under the ECM Oracle Home in SOAHOST1 (where your administration servers resides):

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

   (ORACLE_HOME is the ECM home under *MW_HOME*/ecm.)
2. Run the Oracle WebLogic Scripting Tool (WLST):

   ```
   SOAHOST1>./wlst.sh
   ```

3. Run `connect()` and supply the username, 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 I/PM will use to connect to the BPEL system. It should be a BPEL admin user. CSF credentials are username/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, Oracle I/PM is currently leveraging the standard OWSM CSF map named "oracle.wsm.security". To create a credential, use the `createCred` WLST command:

   ```
   wls:/ecm_domain/serverConfig> createCred(map="oracle.wsm.security",
   key="basic.credential", user="weblogic", password="password_for_credential")
   ```

   The "key" in the command is the "alias," which is used for the 'Credential Alias' property of the BPEL connection definition in the Oracle I/PM 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 SOA system with a central LDAP and single sign-on (SSO) system. See Chapter 11, "Integration with Oracle Identity Management" for details on the sample users created.

---

5. Run the list credentials command to verify that the credential was created:

```
wls:/ecm_domain/serverConfig> listCred(map="oracle.wsm.security",
key="basic.credential")
{map=oracle.wsm.security, key=basic.credential}
Already in Domain Runtime Tree

[Name : weblogic, Description : null, expiry Date : null]
PASSWORD: password_for_credential
```

8.16 Configuring a Workflow Connection

Perform these steps to configure a workflow connection:

1. Log in to the WLS_IPM1 imaging console at http://ECHMHOSTVHN1:16000/imaging.
2. From the navigator pane, under Manage Connections, click the Add icon and then **Create Workflow Connection**. The Workflow Connection Basic Information Page opens.
3. 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 8.15, "Configuring BPEL CSF Credentials."
   c. In the **Provider** field, enter your two SOA 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**.

8.17 Configuring Oracle HTTP Server for the WLS_IPM Managed Servers

To enable Oracle HTTP Server to route to IPM_Cluster, which contains the WLS_IPM1 and WLS_IPM2 managed servers, you must set the `WebLogicCluster` parameter to the list of nodes in the cluster as follows:

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

   ```
   # I/PM Application
   <Location /imaging >
       WebLogicCluster ECMHOST1VHN1:16000,ECMHOST2VHN1:16000
       SetHandler weblogic-handler
       WLProxySSL ON
       WLProxySSLPassThrough ON
   </Location>

   # AXF WS Invocation
   <Location /axf-ws >
       WebLogicCluster ECMHOST1VHN1:16000,ECMHOST2VHN1:16000
       SetHandler weblogic-handler
       WLProxySSL ON
       WLProxySSLPassThrough ON
   </Location>
   ```

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

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

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

8.18 Setting the Frontend HTTP Host and Port

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

1. Log in to Oracle WebLogic Server Administration Console.
2. Go to the Change Center section and click **Lock & Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Clusters**. The Summary of Clusters page opens.
5. Select **IPM_Cluster**.
6. Click the **HTTP** tab.
7. Set the following values:
   - **Frontend Host:** ecm.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 frontend host directive in the cluster take effect.

Note: When HTTPS is enabled in the load balancer and the load balancer terminates SSL (the SOA servers receive only HTTP requests, not HTTPS), as suggested in this guide, the endpoint protocol for webservices is set to `http`. Since the load balancer redirects HTTP to HTTPS, this causes the following exception when testing web services functionality in Oracle Enterprise Manager Fusion Middleware Control:

```
(javax.xml.soap.SOAPException:
oracle.j2ee.ws.saaj.ContentTypeException)
```

To resolve this exception, update the URL endpoint:

In the Enterprise Manager Test Page, check **Edit Endpoint URL**.

Within the endpoint URL page:

- Change `http` to `https`.
- Change the default port number (say 80) to SSL port (say 443).

8.19 Validating Access Through Oracle HTTP Server

Verify URLs to ensure that appropriate routing and failover is working from the HTTP Server to the IPM_Cluster. Perform these steps to verify the URLs:

1. While WLS_IPM2 is running, stop WLS_IPM1 using the Oracle WebLogic Server Administration Console.
2. Access http://WEBHOST1:7777/imaging to verify it is functioning properly. (Please note that you will not be able to retrieve reports or data since the I/PM server is down.)
3. Start WLS_IPM1 from the Oracle WebLogic Server Administration Console.
4. Stop WLS_IPM2 from the Oracle WebLogic Server Administration Console.
5. Access http://WEBHOST1:7777/imaging to verify it is functioning properly.
6. Start WLS_IPM2 from the Oracle WebLogic Server Administration Console.

8.20 Configuring Node Manager for the WLS_IPM Managed Servers

It is assumed that the host names used by the WLS_IPM managed servers as listen address have already been configured for host name verification as explained in Section 7.12, "Configuring Node Manager for the WLS_UCM and WLS_IPM Managed Servers."

At this point, once the WLS servers have been added to the domain, the procedure in Section 9.3.5, "Configuring Managed WLS Servers to Use the Custom Keystores" should be performed so that the servers are configured to use custom key stores.

8.21 Configuring Server Migration for the WLS_IPM Managed Servers

Server migration is required for proper failover of the Oracle I/PM components in the event of failure in any of the ECMHOST1 and ECMHOST2 nodes. See Chapter 10, "Configuring Server Migration" for further details. For Oracle I/PM, use the following values for the variables in that chapter:

- Server names:
  - *WLS_SERVER1*: WLS_IPM1
  - *WLS_SERVER2*: WLS_IPM2
- Host names:
  - *HOST1*: ECMHOST1
  - *HOST2*: ECMHOST2
- Cluster name:
  - *CLUSTER*: IPM_Cluster

8.22 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. Also refer to the *Oracle Database Backup and Recovery Guide* for information on database backup.

Perform these steps to back up the installation at this point:

1. Back up the web tier:

   a. Shut down the instance using `opmnctl`.

   ```
   WEBHOST1> ORACLE_BASE/admin/instance_name/bin/opmnctl stopall
   ```

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

   ```
   WEBHOST1> tar -cvpf BACKUP_LOCATION/web.tar MW_HOME
   ```

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

   ```
   WEBHOST1> tar -cvpf BACKUP_LOCATION/web_instance_name.tar ORACLE_INSTANCE
   ```

   d. Start the instance using `opmnctl`:

   ```
   WEBHOST1> cd ORACLE_BASE/admin/instance_name/bin
   WEBHOST1> 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 to create the backup:

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

9 Setting Up Node Manager

This chapter describes how to configure Node Manager in accordance with the EDG recommendations. It contains the following sections:

9.1 About Setting Up Node Manager

Node Manager enables you to start and stop the administration server and the managed servers.

Process

The procedures described in this chapter must be performed for various components of the enterprise deployment topology outlined in Section 1.7, "Enterprise Deployment Reference Topology." Variables are used in this chapter to distinguish between component-specific items:

- *WLS_SERVER*: this refers to a managed WebLogic 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 9.2, "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 9.3, "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.

9.2 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.

9.3 Enabling Host Name Verification Certificates for Node Manager

Perform these steps to set up SSL for communication between Node Manager and the Administration Server:

- Step 1: Generating Self-Signed Certificates Using the utils.CertGen Utility
- Step 2: Creating an Identity Keystore Using the utils.ImportPrivateKey Utility
- Step 3: Creating a Trust Keystore Using the Keytool Utility
- Step 4: Configuring Node Manager to Use the Custom Keystores
- Step 5: Configuring Managed WLS Servers to Use the Custom Keystores
- Step 6: Changing the Host Name Verification Setting for the Managed Servers

9.3.1 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:

   ```
   HOST> cd WL_HOME/server/bin
   HOST> . ./setWLSEnv.sh
   ```

   Verify that the CLASSPATH environment variable is set:

   ```
   HOST> 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, etc.). In this case, SOAHOST2 uses the cert directory created for SOAHOST1 certificates. Create a user-defined directory for the certificates:

   ```
   HOST> mkdir certs
   ```

3. Change directory to the directory that you just created:

   ```
   HOST> cd certs
   ```

4. Run the utils.CertGen tool from the user-defined directory 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:

   ```
   HOST> java utils.CertGen welcome1 HOST.mycompany.com_cert
   HOST.mycompany.com_key domestic HOST.mycompany.com

   HOST> java utils.CertGen welcome1 VIP.mycompany.com_cert
   VIP.mycompany.com_key domestic VIP.mycompany.com
   ```

9.3.2 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*/aserver/*domain_name*/certs).

> **Note:** The identity store is created (if none exists) when you import a certificate and the corresponding key into the identity store using the utils.ImportPrivateKey utility.

2. 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:

   ```
   HOST> java utils.ImportPrivateKey appIdentityKeyStore.jks welcome1
   appIdentity1 welcome1
   ORACLE_BASE/admin/domain_name/aserver/domain_name/certs/HOST.mycompany.com_
   cert.pem
   ORACLE_BASE/admin/domain_name/aserver/domain_name/certs/HOST.mycompany.com_
   key.pem

   HOST> java utils.ImportPrivateKey appIdentityKeyStore.jks welcome1
   appIdentity2 welcome1
   ORACLE_BASE/admin/domain_name/aserver/domain_name/certs/VIP.mycompany.com_
   cert.pem
   ORACLE_BASE/admin/domain_name/aserver/domain_name/certs/VIP.mycompany.com_
   key.pem
   ```

9.3.3 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 located under the *WL_HOME*/server/lib directory to the same directory as the certificates. For example:

   ```
   HOST> cp WL_HOME/server/lib/cacerts ORACLE_BASE/admin/domain_name/aserver/
   domain_name/certs/appTrustKeyStore.jks
   ```

2. The default password for the standard Java keystore is 'changeit'. Oracle recommends always changing the default password. Use the keytool utility to do this. The syntax is:

   ```
   HOST> keytool -storepasswd -new New_Password -keystore Trust_Keystore
   -storepass Original_Password
   ```

   For example:

   ```
   HOST> keytool -storepasswd -new welcome1 -keystore appTrustKeyStore.jks
   -storepass changeit
   ```

3. The CA certificate CertGenCA.der is used to sign all certificates generated by the utils.CertGen tool. It is located in the *WL_HOME*/server/lib directory. This CA certificate must be imported into the appTrustKeyStore using the keytool utility. The syntax is:

   ```
   HOST> keytool -import -v -noprompt -trustcacerts -alias Alias_Name
   -file CA_File_Location -keystore Keystore_Location -storepass Keystore_Password
   ```

   For example:

   ```
   HOST> keytool -import -v -noprompt -trustcacerts -alias clientCACert -file
   WL_HOME/server/lib/CertGenCA.der -keystore appTrustKeyStore.jks -storepass
   welcome1
   ```

9.3.4 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/aserver/domain_name/
certs/appIdentityKeyStore.jks
CustomIdentityKeyStorePassPhrase=welcome1
CustomIdentityAlias=appIdentity2
CustomIdentityPrivateKeyPassPhrase=welcome1
```

The passphrase entries in the nodemanager.properties file get encrypted when you start Node Manager as described in Section 9.4, "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 9.3.2, "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 before starting Node Manager in the different nodes:

```
HOST> cd WL_HOME/server/bin
HOST> 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.

9.3.5 Configuring Managed WLS Servers to Use the Custom Keystores

Follow these steps to configure the identity and trust keystores for *WLS_SERVER*:

1. Log in to Oracle WebLogic Server Administration Console.
2. Click **Lock & Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page opens.
5. Click the name of the server for which you want to configure the identity and trust keystores (*WLS_SERVER*). The settings page for the selected server opens.
6. Select **Configuration**, 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/aserver/domain_name/certs/
     appIdentityKeyStore.jks
     ```

   - **Custom Identity Keystore Type:** Leave blank; it defaults to JKS.
   - **Custom Identity Keystore Passphrase:** The password (*Keystore_Password*) you provided in Section 9.3.3, "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 in order to write to the keystore. However, some keystores do not require the passphrase to read from the keystore. WebLogic Server only reads 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/aserver/domain_name/certs/
     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 9.3.3, "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 in order to write to the keystore. However, some keystores do not require the passphrase to read from the keystore. WebLogic Server only reads 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 9.3.2, "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.

9.3.6 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 Oracle WebLogic Server Administration Console.
2. Expand the **Environment** node in the Domain Structure window.
3. Click **Servers**. The Summary of Servers page opens.
4. Select the managed server in the Names column of the table. The settings page for the server opens.
5. 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.

9.4 Starting Node Manager

Run these commands to start Node Manager on *HOST*:

> **Note:** If you have not configured and started Node Manager for the first time yet, run the setNMProps.sh script as specified in Section 5.4, "Starting Node Manager on SOAHOST1" to enable the use of the start script for your managed servers.

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

10

Configuring Server Migration

This chapter describes how to configure server migration in accordance with the EDG recommendations. It contains the following sections:

10.1 About Configuring Server Migration

The procedures described in this chapter must be performed for various components of the enterprise deployment topology outlined in Section 1.7, "Enterprise Deployment Reference Topology." Variables are used in this chapter to distinguish between component-specific items:

- *WLS_SERVER1* and *WLS_SERVER2*: these refer to the managed WebLogic servers for the enterprise deployment component
- *HOST1* and *HOST2*: these refer to the host machines for the enterprise deployment component
- *CLUSTER*: this refers to the cluster associated with the enterprise deployment component.

The values to be used to 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. Configuring server migration for the WLS managed servers consists of the following steps:

- Step 1: Setting Up a User and Tablespace for the Server Migration Leasing Table
- Step 2: Creating a Multi-Data Source Using the Oracle WebLogic Server Administration Console
- Step 3: Editing Node Manager's Properties File
- Step 4: Setting Environment and Superuser Privileges for the wlsifconfig.sh Script
- Step 5: Configuring Server Migration Targets
- Step 6: Testing the Server Migration

10.2 Setting Up a User and Tablespace for the Server Migration Leasing Table

The first step is to set up a user and tablespace for the server migration leasing table:

> **Note:** If other servers in the same domain have already been configured with server migration, the same tablespace and data sources can be used. In that case, the data sources and multi-data source for database leasing do not need to be re-created, but they will have to be retargeted to the cluster being configured with server migration.

1. Create a tablespace called 'leasing'. For example, log on 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;
```

2. Create a user named 'leasing' and assign to it the leasing tablespace:

```
SQL> create user leasing identified by welcome1;
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 the *WL_HOME*/server/db/oracle/817 or the *WL_HOME*/server/db/oracle/920 directory to your database node.

   b. Connect to the database as the leasing user.

   c. Run the leasing.ddl script in SQL*Plus:

```
SQL> @Copy_Location/leasing.ddl;
```

10.3 Creating a Multi-Data Source Using the Oracle WebLogic Server Administration Console

The second step is to create a multi-data source for the leasing table from the Oracle WebLogic Server Administration Console. You create a data source to each of the Oracle RAC database instances during the process of setting up the multi-data source, both for these data sources and the global leasing multi-data source.

Please note the following considerations when creating a data source:

- Make sure that this is a non-XA data source.
- The names of the multi-data sources are in the format of *<MultiDS>-rac0*, *<MultiDS>-rac1*, and so on.
- Use Oracle's Driver (Thin) Version 9.0.1, 9.2.0, 10, 11.
- Data sources do not require support for global transactions. Therefore, do *not* use any type of distributed transaction emulation or participation algorithm for the data source (do not choose the **Supports Global Transactions** option, the **Logging Last Resource**, **Emulate Two-Phase Commit**, or **One-Phase Commit** options of the **Supports Global Transactions** option), and specify a service name for your database.
- Target these data sources to the cluster assigned to the enterprise deployment component (*CLUSTER*; see the component-specific chapters in this guide).
- Make sure the initial connection pool capacity of the data sources is set to 0 (zero). To do this, select **Services** and then **Data Sources**. In the Summary of JDBC Data Sources screen, click the data source name in the list, then click the **Connection Pool** tab, and enter **0** (zero) in the **Initial Capacity** field. Click **Save**.

Creating a Multi-Data Source

Perform these steps to create a multi-data source:

1. Log in to the Oracle WebLogic Server Administration Console.
2. In the Domain Structure area in the left, expand the **Services** node and then select the **Data Sources** node. The Summary of JDBC Data Source page opens.
3. Click **Lock & Edit**.
4. In the Summary of JDBC Data Source page, click **New** and choose **Multi Data Source** from the list. The Create a New JDBC Multi Data Source page opens.
5. Enter `leasing` as the name.
6. Enter `jdbc/leasing` as the JNDI name.
7. Select **Failover** as algorithm (default).
8. Click **Next.**
9. Select the cluster assigned to the enterprise deployment component as the target. (See the *CLUSTER* variable in the component-specific chapters in this guide.)
10. Click **Next**.
11. Select **non-XA driver** (the default).
12. Click **Next**.
13. Click **Create New Data Source**.
14. Enter `leasing-rac0` as the name. Enter `jdbc/leasing-rac0` as the JNDI name. Enter `oracle` as the database type. For the driver type, select **Oracle driver (Thin) for RAC Service-Instance connections, Versions: 10 and later**.

> **Note:** When creating the multi-data sources for the leasing table, enter names in the format of <*MultiDS*>-rac0, <*MultiDS*>-rac1, and so on.

15. Click **Next**.
16. Deselect **Supports Global Transactions**.
17. Click **Next**.
18. Enter the service name, database name, host name, host port, database user name, and password for your leasing schema.
19. Click **Next**.
20. Click **Test Configuration** and verify that the connection works.
21. Click **Next**.
22. Target the data source to the cluster assigned to the enterprise deployment component (*CLUSTER*), and click **Finish**.
23. Select the data source and add it to the right screen.
24. Click **Create a New Data Source** for the second instance of your Oracle RAC database, target it to the cluster assigned to the EDG component (*CLUSTER*), repeating the steps for the second instance of your Oracle RAC database.
25. Add the second data source to your multi-data source.
26. Click **Activate Changes**.

10.4 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 steps below are not required if the server properties (start properties) have been properly set and Node Manager can start the servers remotely.

1. 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.
2. Start Node Manager on *HOST1* and *HOST2* by running the startNodeManager.sh script, which is located in the *WL_HOME*/server/bin directory.

Note: When running 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:

```
HOSTn> export JAVA_OPTIONS=-DInterface=eth3
```

and start Node Manager after the variable has been set in the shell.

10.5 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 these files:

Table 10–1 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 sudo configuration 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 sudo privilege to the WebLogic 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 user ('oracle'). The following is an example of an entry inside /etc/sudoers granting 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 rights as appropriate to this step.

10.6 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 Oracle WebLogic Server Administration Console (http://*Host*:*Admin_Port*/console). Typically, *Admin_Port* is 7001 by default.
2. In the Domain Structure window, expand **Environment** and select **Clusters**. The Summary of Clusters page opens.
3. Click the cluster for which you want to configure migration (***CLUSTER***) in the Name column of the table.
4. Click 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 window of the Oracle WebLogic Server Administration Console, expand **Environment** and select **Servers**.

       Tip: Click **Customize this table** in the Summary of Servers page and move Current Machine from the Available window to the Chosen window to view the machine on which the server is running. This will be different from the configuration if the server gets migrated automatically.

    b. Select the server for which you want to configure migration.
    c. Click 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.

10.7 Testing the Server Migration

The sixth and final step is to test the server migration. Perform these steps to verify that server migration is working properly:

From *HOST1*:

1. Stop the *WLS_SERVER1* managed server. To do this, run this command:

   ```
   HOST1> kill -9 pid
   ```

   where *pid* specifies the process ID of the managed server. You can identify the pid in the node by running this command:

   ```
   HOST1> 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 the soa-infra console in the same IP.

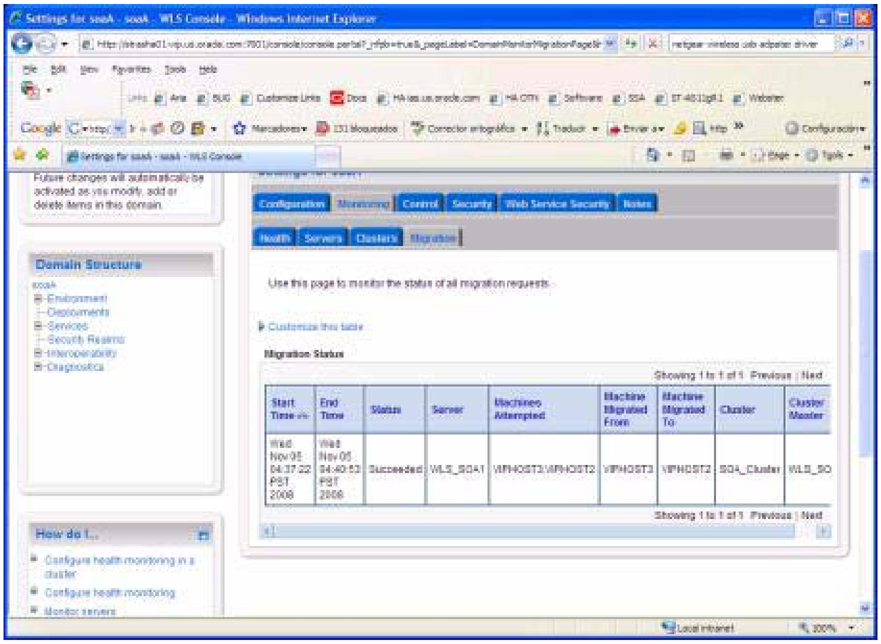
Verification from the Administration Console

Migration can also be verified in the Administration Console:

1. Log in to the Administration Console.
2. Click **Domain** on the left console.
3. Click the **Monitoring** tab and then the **Migration** subtab.

   The Migration Status table provides information on the status of the migration (Figure 10–1).

Figure 10–1 Migration Status Screen in the Administration Console



Use this page to monitor the status of all migration requests.

Migration Status

Showing 1 to 1 of 1 Previous | Next

Start Time	End Time	Status	Server	Machines Attempted	Machine Migrated From	Machine Migrated To	Cluster	Cluster Master
Wed Nov 05 04:37:22 PST 2008	Wed Nov 05 04:40:53 PST 2008	Succeeded	WLS_SOA1	VIPHOST3,VIPHOST2	VIPHOST3	VIPHOST2	SOA_Cluster	WLS_SO

Showing 1 to 1 of 1 Previous | Next

Note: After a server is migrated, to fail it back to its original node or machine, stop the managed server from the Oracle 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.

11

Integration with Oracle Identity Management

This chapter describes how to integrate Oracle Enterprise Content Management Suite with Oracle Identity Management. It contains the following sections:

- Section 11.1, "Credential and Policy Store Configuration"
- Section 11.2, "Oracle Access Manager 10g Integration"
- Section 11.3, "Oracle Access Manager 11g Integration"
- Section 11.4, "Validating Access Through Oracle HTTP Server and SSO"
- Section 11.5, "Backing Up the Installation"

11.1 Credential and Policy Store Configuration

The following topics describe credential and policy store configuration in detail:

- Section 11.1.1, "Overview of Credential and Policy Store Configuration"
- Section 11.1.2, "Credential Store Configuration"
- Section 11.1.3, "Policy Store Configuration"
- Section 11.1.4, "Reassociation of Credentials and Policies"

11.1.1 Overview of Credential and Policy Store Configuration

Oracle Fusion Middleware allows using different types of credential and policy stores in a WebLogic domain. Domains can use stores based on an XML file or on different types of LDAP providers. When a domain uses an LDAP store, all policy and credential data is kept and maintained in a centralized store. However, when using XML policy stores, the changes made on managed servers are not propagated to the Administration Server unless they use the same domain home. The enterprise deployment topology for Oracle Enterprise Content Management Suite 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, Oracle WebLogic Server domains use an XML file for the policy store. The following sections describe the steps required to change the default store to Oracle Internet Directory LDAP for credentials or policies.

Note: The backend repository for the policy store and the credential store must use the same kind of LDAP server. To preserve this coherence, note that reassociating one store implies reassociating the other one, that is, the reassociation of both the credential and the policy stores is accomplished as a unit using the Fusion Middleware Control or the WLST command `reassociateSecurityStore`. For more information, see Section 11.1.4, "Reassociation of Credentials and Policies."

11.1.2 Credential Store Configuration

A credential store is a repository of security data (credentials). A credential can hold user name and password combinations, tickets, or public key certificates. Credentials are used during authentication, when principals are populated in subjects, and, further, during authorization, when determining what actions the subject can perform. This section provides steps to configure Oracle Internet Directory LDAP as a credential store for the Oracle Enterprise Content Management Suite enterprise deployment topology. For more details on credential store configuration, refer to the "Configuring the Credential Store" chapter in the *Oracle Fusion Middleware Security Guide*.

The following section describe credential store configuration:

- Section 11.1.2.1, "Creating the LDAP Authenticator"
- Section 11.1.2.2, "Moving the WebLogic Administrator to LDAP"
- Section 11.1.2.3, "Reassociating the Domain Credential Store"

11.1.2.1 Creating the LDAP Authenticator

To be safe, before you create the LDAP authenticator, you should first back up the relevant configuration files:

```
ORACLE_BASE/admin/domain_name/aserver/domain_name/config/config.xml
ORACLE_BASE/admin/domain_name/aserver/domain_name/config/fmwconfig/jps-config.xml
ORACLE_BASE/admin/domain_name/aserver/domain_name/config/fmwconfig/
system-jazn-data.xml
```

Also back up the boot properties file for the Administration Server:

```
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 WebLogic Server Console.
2. Click the **Security Realms** link on the left navigational bar.
3. Click the **myrealm** default realm entry to configure it.
4. Open the **Providers** tab within the realm.
5. Observe that there is a `DefaultAuthenticator` provider configured for the realm.
6. Click **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.

11.1.2.1.1 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**

11.1.2.2 Moving the WebLogic Administrator to LDAP

This section provides details for provisioning a new administrator user and group for managing Oracle Enterprise Content Management Suite's WebLogic domain in the enterprise deployment topology. This section describes the following tasks:

- Section 11.1.2.2.1, "Provisioning Admin Users and Groups in an LDAP Directory"
- Section 11.1.2.2.2, "Assigning the Admin Role to the Admin Group"
- Section 11.1.2.2.3, "Updating the boot.properties File and Restarting the System"

11.1.2.2.1 Provisioning Admin Users and Groups in an LDAP Directory As mentioned in the introduction to this section, users and groups from multiple WebLogic domains may be provisioned in a central LDAP user store. In such a case, there is a possibility that one WebLogic admin user may have access to all the domains within an enterprise. This is not a desirable situation. To avoid this, the users and groups provisioned must have a unique distinguished name within the directory tree. In this guide, the admin user and group for the ECM EDG WebLogic domain will be provisioned with the DNs below:

- 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 contents shown below depending on the OAM version used, and then save the file:

   - **OAM 10*g*:**

     ```
     dn: cn=weblogic_ecm, cn=Users, dc=us, dc=mycompany, dc=com
     orclsamaccountname: weblogic_ecm
     givenname: weblogic_ecm
     sn: weblogic_ecm
     userpassword: Welcome1
     obver: 10.1.4.0
     mail: weblogic_ecm
     objectclass: top
     objectclass: person
     objectclass: organizationalPerson
     objectclass: inetorgperson
     objectclass: orcluser
     objectclass: orcluserV2
     objectclass: oblixorgperson
     uid: weblogic_ecm
     cn: weblogic_ecm
     description: Admin User for the ECM Domain
     ```

   - **OAM 11*g*:**

     ```
     dn: cn=weblogic_ecm, cn=Users, dc=us, dc=mycompany, dc=com
     orclsamaccountname: weblogic_ecm
     givenname: weblogic_ecm
     sn: weblogic_ecm
     userpassword: Welcome1
     obver: 10.1.4.0
     ```

```
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 ECM Domain
```

2. Run the `ldapadd` command located under the *ORACLE_HOME*/bin directory to provision the user in Oracle Internet Directory.

> **Note:** The ORACLE_HOME used here is the ORACLE_HOME for the Identity Management installation where Oracle Internet Directory resides.

   For example (the command is shown as two lines in the example below for readability purposes, but you should enter the command on a single line):

```
OIDHOST1> ORACLE_HOME/bin/ldapadd -h oid.mycompany.com -p 389 -D
cn="orcladmin" -w welcome1 -c -v -f admin_user.ldif
```

3. Create an ldif file named admin_group.ldif with the contents shown below and then save the file:

```
dn: cn=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 ECM Domain
```

4. Run the `ldapadd` command located under the *ORACLE_HOME*/bin directory to provision the group in Oracle Internet Directory (the command is shown as two lines in the example below for readability purposes, but you should enter the command on a single line):

```
OIDHOST1> ORACLE_HOME/bin/ldapadd -h oid.mycompany.com -p 389 -D
cn="orcladmin" -w welcome1 -c -v -f admin_group.ldif
```

11.1.2.2.2 Assigning the Admin Role to the Admin Group After adding the users and groups to Oracle Internet Directory, the group must be assigned the Admin role within the WebLogic domain security realm. This enables all users that belong to the group to be administrators for that domain. Follow these steps to assign the Admin role to the Admin group:

1. Log 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, click the Roles & Policies tab.

5. On the Realm Roles page, expand the Global Roles entry under the Roles table. This brings up the entry for Roles. Click on the **Roles** link to bring up the Global Roles page.
6. On the Global Roles page, click the Admin role to bring up the Edit Global Role page:
   a. On the Edit Global Roles page, under the Role Conditions table, click the **Add Conditions** button.
   b. On the Choose a Predicate page, select **Group** from the drop down list for predicates and click **Next**.
   c. On the Edit Arguments Page, specify **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 Administration Server Console using a web browser. Log in using the credentials for the weblogic_ecm user.

> **Note:** Each Oracle application in the Oracle ECM 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 SOA administrators are also administrators for the Oracle WebLogic Server domain where Oracle SOA, I/PM and UCM are running. This is why it may be desirable, as suggested in this section, to create a more specific group such as "ECM Administrators." In order 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.

11.1.2.2.3 Updating the boot.properties File and Restarting the System The boot.properties file for the Administration Server should be updated with the WebLogic admin user created in Oracle Internet Directory. Follow the steps below to update the boot.properties file:

1. On SOAHOST1, go the following directory:

```
SOAHOST1>cd ORACLE_BASE/admin/domain_name/aserver/domain_name/servers/
AdminServer/security
```

2. Rename the existing boot.properties file:

```
SOAHOST1> mv boot.properties boot.properties.backup
```

3. Use a text editor to create a file called boot.properties under the security directory. Enter the following lines in the file:

```
username=weblogic_ecm
password=welcome1
```

4. Save the file.
5. Stop the Administration Server:

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

6. Start the Administrator Server using the procedure in Section 5.5, "Starting the Administration Server on SOAHOST1."

11.1.2.3 Reassociating the Domain Credential Store

The reassociation of both the credential and the policy stores is accomplished as a unit using Fusion Middleware Control or the WLST command `reassociateSecurityStore`. See Section 11.1.4, "Reassociation of Credentials and Policies" for detailed steps.

11.1.3 Policy Store Configuration

The domain policy store is the repository of system and application-specific policies. In a given domain, there is one store that stores all policies that all applications deployed in the domain may use. This section provides the steps to configure Oracle Internet Directory LDAP as the policy store for the Oracle Enterprise Content Management Suite enterprise deployment topology. For more details on policy store configuration, refer to the "OPSS Authorization and the Policy Store" chapter in the *Oracle Fusion Middleware Security Guide*.

11.1.3.1 Prerequisites to Using an LDAP-Based Policy Store

In order to ensure the proper access to an LDAP server directory (Oracle Internet Directory) used as a policy store, you must set a node in the server directory.

An Oracle Internet Directory administrator must follow these steps to create the appropriate node in an Oracle Internet Directory Server:

1. Create an LDIF file (assumed to be `jpstestnode.ldif` in this example) specifying the following DN and CN entries:

   ```
   dn: cn=jpsroot_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 domains. It is not required that this node be created at the top level, as long as read and write access to the subtree is granted to the Oracle Internet Directory administrator.

2. Import this data into Oracle Internet Directory server using the `ldapadd` command, as illustrated in the following example (the command is shown as two lines in the example below for readability purposes, but you should enter the command on a single line):

   ```
   OIDHOST1> ORACLE_HOME/bin/ldapadd -h ldap_host -p ldap_port -D
   cn=orcladmin -w password -c -v -f jpstestnode.ldif
   ```

3. Verify that the node has been successfully inserted using the `ldapsearch` command, as illustrated in the following example (the command is shown as two lines in the example below for readability purposes, but you should enter the command on a single line):

```
OIDHOST1> ORACLE_HOME/bin/ldapsearch -h ldap_host -p ldap_port -D
cn=orcladmin -w password -b "cn=jpsroot_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 sqlplus, enter the command to gather the statistics info:

   ```
   SQLPLUS> @ORACLE_HOME/ldap/admin/oidstats.sql
   ```

   The oidstats.sql utility must be run just once after the initial provisioning. For details about this utility, consult the *Oracle Fusion Middleware User Reference for Oracle Identity Management*.

11.1.3.2 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 Enterprise Manager Fusion Middleware Control or the WLST command `reassociateSecurityStore`. See Section 11.1.4, "Reassociation of Credentials and Policies" for detailed steps.

11.1.4 Reassociation of Credentials and Policies

To reassociate the policy and credential store with Oracle Internet Directory, use the WLST `reassociateSecurityStore` command. Follow these steps:

1. From SOAHOST1, start the `wlst` shell:

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

2. Connect to the WebLogic Administration Server using the `wlst connect` command shown below:

   Syntax:

   ```
   connect('Admin_User',"Admin_User_Password",t3://hostname:port)
   ```

   For example:

   ```
   connect("weblogic","welcome1","t3://ADMINVHN:7001")
   ```

3. Run the `reassociateSecurityStore` command as shown below:

   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="welcome1",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=welcome1}
Location changed to domainRuntime tree. This is a read-only tree with
DomainMBean as the root.
For more help, use help(domainRuntime)

Starting Policy Store reassociation.
LDAP server and ServiceConfigurator setup done.

Schema is seeded into LDAP server
Data is migrated to LDAP server
Service in 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 5.5, "Starting the Administration Server on SOAHOST1."

> **Note:** For credential and policy changes to take effect, the servers in the domain must be restarted.

11.1.4.1 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 *Oracle Fusion Middleware Reference for Oracle Identity Management*.

11.2 Oracle Access Manager 10*g* Integration

This section describes how to set up Oracle Access Manager 10*g* as the single sign-on solution for the Oracle Enterprise Content Management Suite enterprise deployment topology. It contains the following sections:

- Section 11.2.1, "Overview of Oracle Access Manager Integration"
- Section 11.2.2, "Prerequisites for Oracle Access Manager"
- Section 11.2.3, "Configuring Oracle Access Manager"
- Section 11.2.4, "Installing and Configuring WebGate"
- Section 11.2.5, "Configuring IP Validation for the EDG Webgate"
- Section 11.2.6, "Setting Up WebLogic Authenticators"

11.2.1 Overview of Oracle Access Manager Integration

Oracle Access Manager (OAM) is the recommended single sign-on (SSO) solution for Oracle Fusion Middleware 11*g* Release 1. For more information on installing and configuring an OAM installation, see *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*. This section explains the procedure for configuring the Oracle ECM installation with an existing OAM 10*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 ECM enterprise deployment topology described in this book uses a single sign-on configuration where both the Oracle ECM system and the single sign-on system are in the same network domain (mycompany.com). For a multi-domain configuration, please refer to the required configuration steps in "Configuring Single Sign-On" of the *Oracle Access Manager Access Administration Guide*.

11.2.2 Prerequisites for Oracle Access Manager

The setup for Oracle Access Manager (OAM) assumes an existing OAM 10*g* installation complete with Access Managers and a policy protecting the Policy Manager. For more information on installing and configuring an OAM installation, see *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*. This setup includes a directory service such as Oracle Internet Directory (OID), either stand-alone or as part of an Oracle Virtual Directory (OVD) configuration. This section provides the necessary steps for configuring your enterprise deployment with either OID or OVD.

In addition, the OAM installation should have its own Web server configured with WebGate. This section also provides the steps for using the OAM Web server as a delegated authentication server.

11.2.3 Configuring Oracle Access Manager

This section covers the following topics:

- Section 11.2.3.1, "Collecting the Information for the OAM Configuration Tool"
- Section 11.2.3.2, "Running the OAM Configuration Tool"
- Section 11.2.3.3, "Oracle Access Manager Logout Guidelines"
- Section 11.2.3.4, "Verifying Successful Creation of the Policy Domain and AccessGate"
- Section 11.2.3.5, "Verifying That the Cookieless Basic Authorization Scheme Has Been Properly Assigned"
- Section 11.2.3.6, "Updating the Host Identifier"
- Section 11.2.3.7, "Updating the WebGate Profile"
- Section 11.2.3.8, "Adding Additional Access Servers"
- Section 11.2.3.9, "Configuring Delegated Form Authentication"

11.2.3.1 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 OAM
2. Policies to enable authentication in Oracle WebLogic Server
3. A WebGate entry in OAM to enable Oracle HTTP Server WebGates (from your Web Tier) to protect your configured application
4. A host identifier, depending on the scenario chosen (a default host identifier would be used, if not provided)
5. 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.

11.2.3.2 Running the OAM Configuration Tool

Before running the OAM Configuration Tool, you must first add the required resources to OAM 10*g* for the Oracle ECM components. Create a file containing the list of URIs you want to protect with the following content:

```
########################
#Product Name: ECM
########################
###########################
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 OAM 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. Refer to the *Oracle Access Manager Access Administration Guide* for details on the different patterns that can be used if more specialized protection patterns need to be used.

The OAM Configuration tool resides in the *ORACLE_COMMON_HOME*/ modules/oracle.oamprovider_11.1.1 directory (*ORACLE_COMMON_HOME* depends on which machine 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 ECMHOST1. Run the OAM Configuration Tool for OAM 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
```

11.2.3.3 Oracle Access Manager Logout Guidelines

For applications invoked by Oracle UCM and Oracle I/PM 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 *Oracle Fusion Middleware Security Guide*.

When integrating Oracle UCM with OAM 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. See "Configuring a Single Sign-On Logout URL" and "AccessGate Configuration Parameters" in the *Oracle Access Manager Access Administration Guide* for more information.

11.2.3.4 Verifying Successful Creation of the Policy Domain and AccessGate

Verifying the Policy Domain

Perform these steps 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. Click 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

Perform these steps 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. Click 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 11.2.3.2, "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.

11.2.3.5 Verifying That the Cookieless Basic Authorization Scheme Has Been Properly Assigned

Perform these steps 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.

11.2.3.6 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.

Perform these steps 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 username and password, log in as an administrator. Click **OK**.

3. On the Access System main page, click the **Access System Console** link.

4. On the Access System Console page, 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 on 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
   ecmhost1vhn1.mycompany.com:16000
   ecmhost2vhn1.mycompany.com:16000
   ecmhost1.mycompany.com:16200
   ecmhost2.mycompany.com:16200
   ecm.mycompany.com:443
   ecminternal.mycompany.com:80
   admin.mycompany.com:80
   adminvhn.mycompany.com:7001
   sso.mycompany.com:7779 [WebGate access with Oracle IDM port]
   ```

9. Select the check box next to Update Cache and then click **Save**.

   A message box with the following message is displayed: "Updating the cache at this point will flush all the caches in the system. Are you sure?".

   Click **OK** to finish saving the configuration changes.

10. Verify the changes on the Host Identifier Details page.

11.2.3.7 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.

Perform these steps 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 hostname with the name of the computer where WebGate is running, for example: `webhost1.mycompany.com`.
   - **Preferred HTTP Host**: Update the Preferred_HTTP_Host with one of the hostname variations specified in the previous section, for example: `admin.mycompany.com: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 "Are you sure you want to commit these changes?" message is displayed.

   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.

11.2.3.8 Adding Additional Access Servers

Perform these steps 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.

11.2.3.9 Configuring Delegated Form Authentication

Perform these steps to configure the form authentication to redirect to the WebGate that was installed with the OAM 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 IDM installation; for example: `http://sso.mycompany.com`. Click **Save** when you are done.

A WebGate should already be installed in the IDM installation. Refer to *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management* for details.

11.2.4 Installing and Configuring WebGate

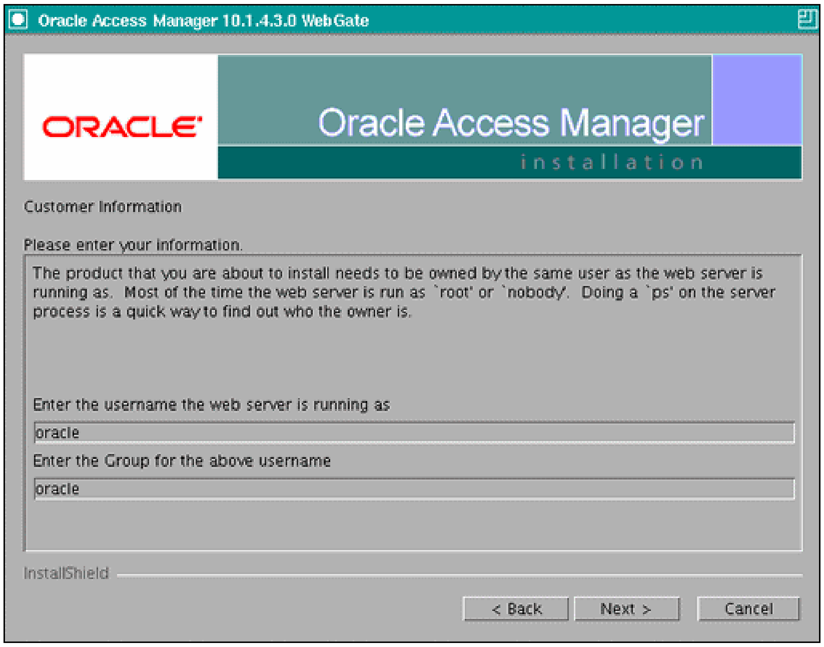
WebGate needs to be installed on each of the WEBHOST*n* machines in order to secure the web tier:

> **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 1.8, "What to Install" for information on where to obtain it) using the following command:

   ```
   WEBHOSTn> ./Oracle_Access_Manager10_1_4_3_0_linux_OHS11g_WebGate -gui
   ```
2. The Welcome screen opens. Click **Next**.
3. In the Customer Information screen (Figure 11–1), enter the user name and user group that the web server is running as. Click **Next** to continue.

Figure 11–1 Customer Information Screen



Customer Information

Please enter your information.

The product that you are about to install needs to be owned by the same user as the web server is running as. Most of the time the web server is run as `root' or `nobody'. Doing a `ps' on the server process is a quick way to find out who the owner is.

Enter the username the web server is running as

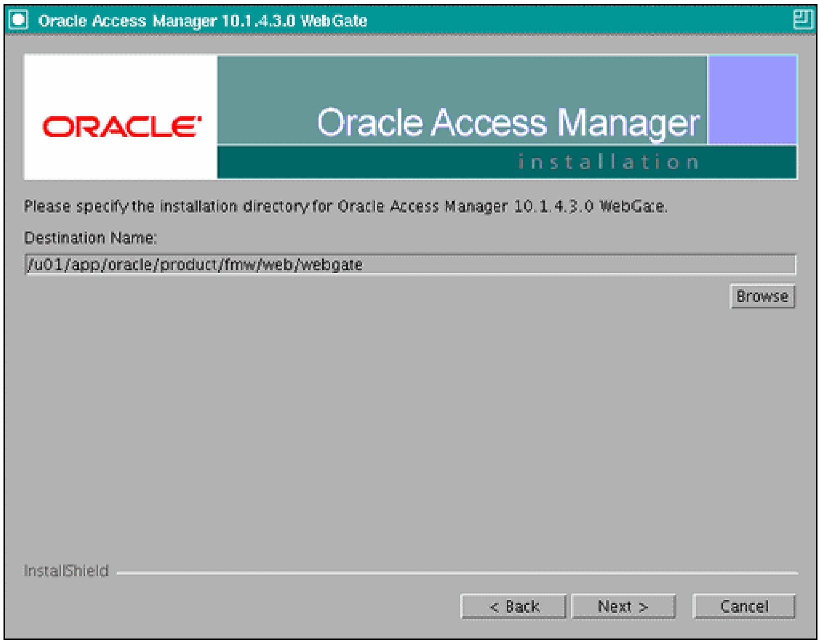
oracle

Enter the Group for the above username

oracle

4. In the installation target screen (Figure 11–2), specify the directory where WebGate should be installed. Click **Next** to continue.

Figure 11–2 Installation Target Screen



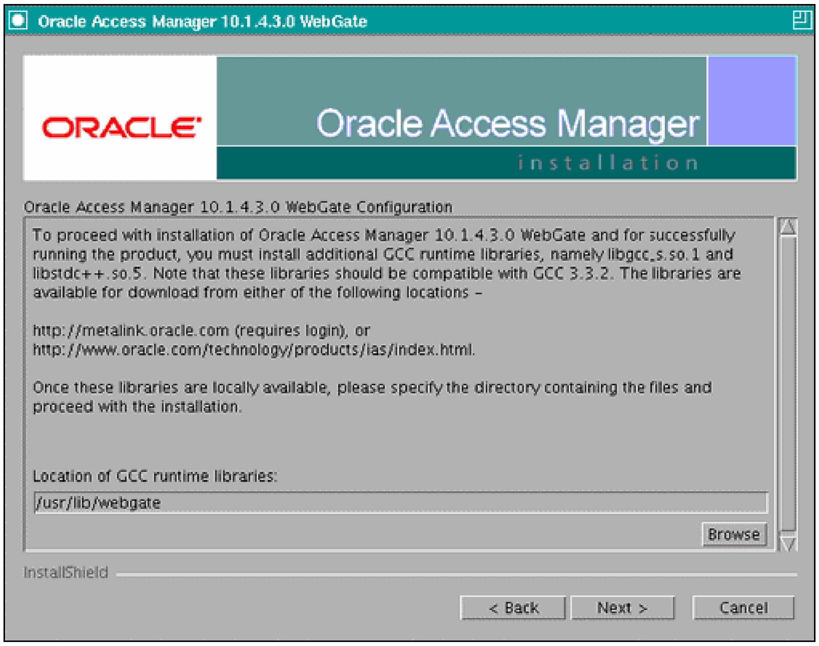
Please specify the installation directory for Oracle Access Manager 10.1.4.3.0 WebGate.

Destination Name:

/u01/app/oracle/product/fmw/web/webgate

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 11–3), and use **Browse** to point to their location on the local computer. Click **Next** to continue.

Figure 11–3 Runtime Libraries Screen



Oracle Access Manager 10.1.4.3.0 WebGate Configuration

To proceed with installation of Oracle Access Manager 10.1.4.3.0 WebGate and for successfully running the product, you must install additional GCC runtime libraries, namely libgcc_s.so.1 and libstdc++.so.5. Note that these libraries should be compatible with GCC 3.3.2. The libraries are available for download from either of the following locations -

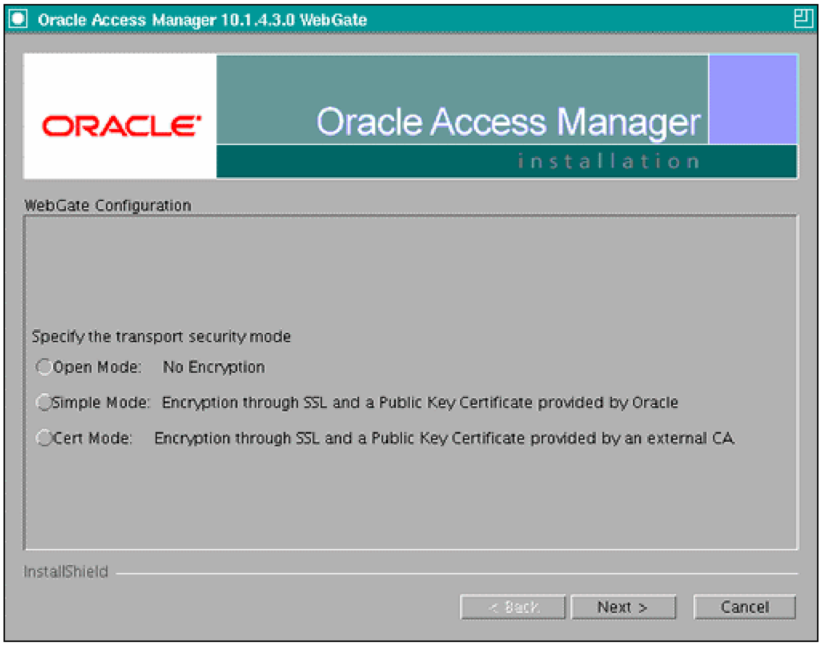
http://metalink.oracle.com (requires login), or
http://www.oracle.com/technology/products/ias/index.html.

Once these libraries are locally available, please specify the directory containing the files and proceed with the installation.

Location of GCC runtime libraries:
/usr/lib/webgate

7. The installer now creates the required artifacts. After that is completed, click **Next** to continue.
8. In the transport security mode screen (Figure 11–4), select "Open Mode: No Encryption" and click **Next** to continue.

Figure 11–4 Transport Security Mode Screen



WebGate Configuration

Specify the transport security mode

○ Open Mode: No Encryption

○ Simple Mode: Encryption through SSL and a Public Key Certificate provided by Oracle

○ Cert Mode: Encryption through SSL and a Public Key Certificate provided by an external CA

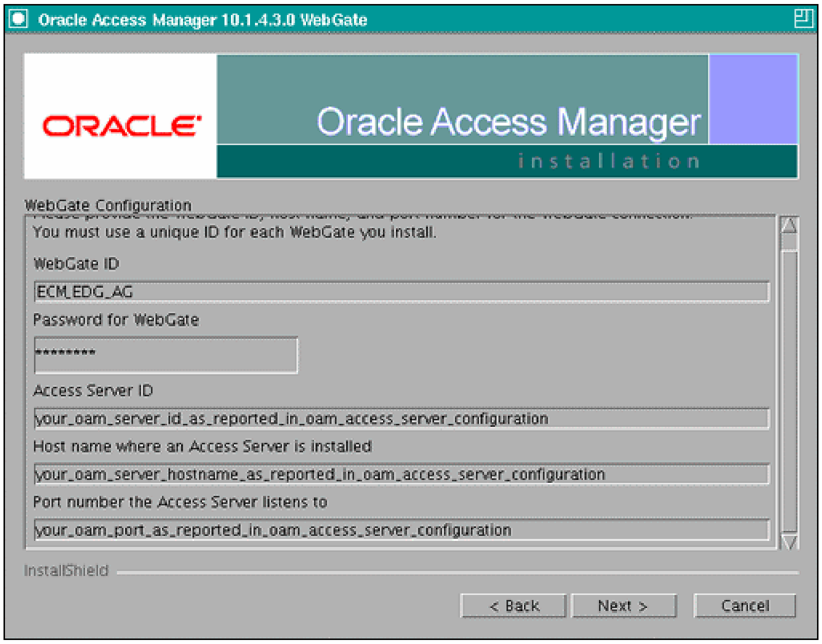
9. In the WebGate configuration screen, provide the details of the access server that will be used. You must provide the following information:

   - **WebGate ID**, as provided when the OAM configuration tool was executed
   - **Password for WebGate**
   - **Access Server ID**, as reported by the OAM Access Server configuration
   - **Access Server host name**, as reported by the OAM Access Server configuration
   - **Access Server port number**, as reported by the OAM Access Server configuration

   Note: The Access Server ID, host name, and port are all required.

   You can obtain these details from your Oracle Access Manager administrator. Click **Next** to continue.

Figure 11–5 Access Server Configuration Screen



WebGate Configuration

Please provide the WebGate ID, host name, and port number for the WebGate connection.
You must use a unique ID for each WebGate you install.

WebGate ID

ECM_EDG_AG

Password for WebGate

****

Access Server ID

your_oam_server_id_as_reported_in_oam_access_server_configuration

Host name where an Access Server is installed

your_oam_server_hostname_as_reported_in_oam_access_server_configuration

Port number the Access Server listens to

your_oam_port_as_reported_in_oam_access_server_configuration

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.

11.2.5 Configuring IP Validation for the EDG 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.

Perform these steps 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 Oracle Access Manager 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.

11.2.6 Setting Up WebLogic Authenticators

This section assumes that you have already set up the LDAP authenticator by following the steps in Section 11.1.2.1, "Creating the LDAP Authenticator." If you have not already created the LDAP authenticator, do it before continuing with this section.

This section covers the following topics:

- Section 11.2.6.1, "Back Up Configuration Files"
- Section 11.2.6.2, "Setting Up the OAM ID Asserter"
- Section 11.2.6.3, "Setting the Order of Providers"

11.2.6.1 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.

11.2.6.2 Setting Up the OAM ID Asserter

Perform these steps to set up the OAM ID Asserter:

1. Log in to Weblogic 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 **OAM Identity Asserter** 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 OAM 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 OAM 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.

11.2.6.3 Setting the Order of Providers

Reorder the OAM Identity Asserter, OID/OVD Authenticator, and Default Authenticator by ensuring that the control flag for each authenticator is set as follows:

- OAM Identity Asserter: REQUIRED
- OID LDAP Authenticator (or OVD LDAP Authenticator): SUFFICIENT
- Default Authenticator: SUFFICIENT
- 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 8.15, "Configuring BPEL CSF Credentials" for further details. (This book uses the 'weblogic_ecm' user as an example for SSO.)

11.3 Oracle Access Manager 11*g* Integration

This section describes how to set up Oracle Access Manager 11*g* as the single sign-on solution for the Oracle Enterprise Content Management Suite enterprise deployment topology. It contains the following sections:

- Section 11.3.1, "Overview of Oracle Access Manager Integration"
- Section 11.3.2, "Prerequisites for Oracle Access Manager"
- Section 11.3.3, "Setting Up WebGate"
- Section 11.3.4, "Registering the WebGate Agent"
- Section 11.3.5, "Setting Up the WebLogic Authenticators"

11.3.1 Overview of Oracle Access Manager Integration

Oracle Access Manager (OAM) is the recommended single sign-on solution for Oracle Fusion Middleware 11*g* Release 1. For more information on installing and configuring an OAM installation, see *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*. This section explains the procedure for configuring the Oracle ECM installation with an existing OAM 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 ECM enterprise deployment topology described in this guide uses a single sign-on configuration where both the Oracle ECM system and the single sign-on system are in the same network domain (mycompany.com). For a multi-domain configuration, please refer to the required configuration steps in "Configuring Single Sign-On," of the *Oracle Access Manager Access Administration Guide*.

11.3.2 Prerequisites for Oracle Access Manager

The setup for Oracle Access Manager (OAM) assumes an existing OAM 11*g* installation complete with Access Managers and a policy protecting the Policy Manager. For more information on installing and configuring an OAM installation, see *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*. This setup includes a directory service such as Oracle Internet Directory (OID), either stand-alone or as part of an Oracle Virtual Directory (OVD) configuration. This section provides the necessary steps for configuring your enterprise deployment with either OID or OVD.

In addition, the OAM installation should have its own Web server configured with WebGate. This section also provides the steps for using the OAM Web server as a delegated authentication server.

11.3.3 Setting Up WebGate

You must set up a WebGate on each of the WEBHOST machines where Oracle HTTP Server has already been installed. Sections 11.3.3 and 11.3.4 should be repeated for each WEBHOST*n* in the deployment environment.

This section covers the following topics:

- Section 11.3.3.1, "Installing GCC Libraries"
- Section 11.3.3.2, "Installing WebGate"
- Section 11.3.3.3, "Post-Installation Steps"

11.3.3.1 Installing GCC Libraries

You must download and install third-party GCC libraries on your machine 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 11–1 lists the versions of third-party GCC libraries for Linux and Solaris.

Table 11–1 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

11.3.3.2 Installing WebGate

This section describes the procedures for installing WebGate.

Launching the Installer

The installer program for Oracle HTTP Server 11*g* Webgate for Oracle Access Manager is included in the webgate.zip file.

Perform these steps 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. Start the installer using the following command:

   ```
   $ ./runInstaller -jreLoc WebTier_Home/jdk
   ```

> **Note:** When you install Oracle HTTP Server, the jdk directory is created under the *WebTier_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.

Perform these steps 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 Middleware Home and Oracle Home locations.

   Note: The Middleware Home contains an Oracle Home for Oracle Web Tier.

   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 11.3.3.1, "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.

11.3.3.3 Post-Installation Steps

Complete the following procedure 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 as the Oracle Home for Webgate, for example:

   ```
   MW_HOME/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_Home directory to the Webgate Instance location:

   ```
   $ ./deployWebGateInstance.sh -w Webgate_Instance_Directory
   -oh Webgate_Oracle_Home
   ```

Webgate_Instance_Directory is the location of Webgate Instance Home, which is same as the Instance Home of Oracle HTTP Server, for example:

```
MW_HOME/Oracle_WT1/instances/instance1/config/OHS/ohs1
```

> **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_Home_for_Oracle_HTTP_
Server/lib
```

4. From your present working directory, move up one directory level:

```
$ cd Webgate_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_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 Webgate_Instance_Directory [-oh Webgate_Oracle_Home]
[-o output_file]
```

> **Note:** The -oh *WebGate_Oracle_Home* and -o *output_file* parameters are optional.

where *WebGate_Oracle_Home* is the directory where you have installed Oracle HTTP Server Webgate for Oracle Access Manager and created as the Oracle Home for Webgate, for example:

```
MW_HOME/Oracle_OAMWebGate1
```

The *Webgate_Instance_Directory* is the location of Webgate Instance Home, which is same as the Instance Home of Oracle HTTP Server, for example:

```
MW_HOME/Oracle_WT1/instances/instance1/config/OHS/ohs1
```

The *output_file* is the name of the temporary output file used by the tool, for example:

```
Edithttpconf.log
```

11.3.4 Registering the WebGate Agent

This section describes the procedures for registering the WebGate Agent:

- Section 11.3.4.1, "Extracting and Using the RREG Tool"
- Section 11.3.4.2, "Updating the OAM11gRequest File"
- Section 11.3.4.3, "Running the oamreg Tool"
- Section 11.3.4.4, "Changing the inspection.wsil Resource to Use the Basic Authentication Scheme"
- Section 11.3.4.5, "Updating the OAM 11g Server configuration to Support the Basic Cookieless Scheme"
- Section 11.3.4.6, "Copying Access files to WEBHOSTs"

11.3.4.1 Extracting and Using the RREG Tool

The RREG tool is part of the OAM 11*g* installation. If it is not already available, extract it using 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.

11.3.4.2 Updating the OAM11gRequest File

The *RREG_Home*/input directory contains a template file named OAM11gRequest.xml. Copy this file to ECMOAM11gRequest.xml and edit that file in order to create the policies for the Oracle ECM installation. After editing, the file should look as follows:

> **Note:** Replace *OAM_HOST*, *OAM_ADMINSERVER_PORT*, *WEBTIER_HOST_ECM*, and *FRONT_END_URL_ECM* with their respective values in your installation.

```
<?xml version="1.0" encoding="UTF-8"?>

<!-- Copyright (c) 2009, 2010, Oracle and/or its affiliates. All rights reserved.

 NAME: OAM11GRequest_short.xml - Template for OAM 11G Agent Registration request
   file
   (Shorter version - Only mandatory values - Default values will be used for all
   other fields)
   DESCRIPTION: Modify with specific values and pass file as input to the tool.
-->

-->
<OAM11GRegRequest>
    <serverAddress>http://OAM_HOST:OAM_ADMINSERVER_PORT</serverAddress>
    <hostIdentifier>WEBTIER_HOST_ECM</hostIdentifier>
    <agentName>WEBTIER_HOST_ECM</agentName>
    <applicationDomain>FRONT_END_URL_ECM</applicationDomain>
    <cachePragmaHeader>private</cachePragmaHeader>
    <cacheControlHeader>private</cacheControlHeader>
    <ipValidation>1</ipValidation>
    <logOutUrls>
        <url>/oamsso/logout.html</url>
```

```
    </logOutUrls>

    <protectedResourcesList>
        <resource>/adfAuthentication</resource>
        <resource>/adfAuthentication/.../*</resource>
        <resource>/imaging/faces</resource>
        <resource>/imaging/faces/.../*</resource>
        <resource>/em</resource>
        <resource>/em/.../*</resource>
        <resource>/console</resource>
        <resource>/console/.../*</resource>
        <resource>/DefaultToDoTaskFlow</resource>
        <resource>/DefaultToDoTaskFlow/.../*</resource>
        <resource>/sdpmessaging/userprefs-ui</resource>
        <resource>/sdpmessaging/userprefs-ui/.../*</resource>
        <resource>/integration/worklistapp</resource>
        <resource>/integration/worklistapp/.../*</resource>
        <resource>/workflow/sdpmessagingsca-ui-worklist</resource>
        <resource>/workflow/sdpmessagingsca-ui-worklist/.../*</resource>
        <resource>/soa/composer</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>/cs/.../*</resource>
        <resource>/_ocsh</resource>
        <resource>/_ocsh/.../*</resource>
        <resource>/imaging</resource>
        <resource>/imaging/.../*</resource>
        <resource>/soa-infra/services/.../*</resource>
        <resource>/soa-infra/directWSDL</resource>
        <resource>/soa-infra/directWSDL/.../*</resource>
        <resource>/ucs/messaging/webservice</resource>
        <resource>/ucs/messaging/webservice/.../*</resource>
    </publicResourcesList>

    <userDefinedParameters>
        <userDefinedParam>
            <name>ipValidationExceptions</name>
            <value>10.1.1.1</value>
        </userDefinedParam>
    </userDefinedParameters>
</OAM11GRegRequest>
```

11.3.4.3 Running the oamreg Tool

Run the oamreg tool using the following command:

```
$ ./RREG_Home/bin/oamreg.sh inband input/ECMOAM11gRequest.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/ECMOAM11GRequest.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:WEBTIER_HOST_ECM
URL String:WEBTIER_HOST_ECM
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.
```

11.3.4.4 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 I/PM to work, this resource must be updated to use the basic authentication scheme instead:

1. Log on 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 (*FRONT_END_URL_ECM*).
3. Expand your application domain's name.
4. Expand **Authentication Policies**.
5. Double-click on **Protected Resource Policy**.
6. Select the inspection.wsil and inspection.wsil/.../* resources and click the **Delete** icon in the Resources pane to remove them. (Please note that the inspection.wsil/.../* resource was automatically added during the registration.)
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 on it and make sure the inspection.wsil and and inspection.wsil/.../* resources were added).

> **Note:** Do not forget to create a new credential for the new user. See Section 8.15, "Configuring BPEL CSF Credentials" for further details. (This book uses the 'weblogic_ecm' user as an example for SSO.)

11.3.4.5 Updating the OAM 11*g* Server configuration to Support the Basic Cookieless Scheme

You must set the `NoUniqueSessionsFor10gAgents` parameter in the OAM 11*g* configuration to ’true’. 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 identity management system for the change to take effect.

11.3.4.6 Copying Access files to WEBHOSTs

The following two files are generated in *RREG_Home*/output/*WEBTIER_HOST_ECM*:

- ObAccessClient.xml
- cwallet.sso

Copy these files to the webgate instance location on the WEBHOST machine (*Webgate_Instance_Home*/config/OHS/ohs*N*/webgate/config/).

11.3.5 Setting Up the WebLogic Authenticators

This section assumes that you have already set up the LDAP authenticator by following the steps in Section 11.1.2.1, "Creating the LDAP Authenticator." If you have not already created the LDAP authenticator, do it before continuing with this section.

This section covers the following topics:

- Section 11.3.5.1, "Backing Up Configuration Files"
- Section 11.3.5.2, "Setting Up the OAM ID Asserter"
- Section 11.3.5.3, "Setting the Order of Providers"

11.3.5.1 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.

11.3.5.2 Setting Up the OAM ID Asserter

Perform these steps to set up the OAM ID Asserter:

1. Log in to Weblogic 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 **OAM Identity Asserter** 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 OAM Identity Asserter.
7. Set the control flag to ’REQUIRED’.
8. Select both the **ObSSOCookie** and **OAM_REMOTE_USER** options under active types.
9. Save the settings.

Finally, log in to the WLST console as an administrator and run the following command:

```
addOAMSSOProvider(loginuri="/${app.context}/adfAuthentication",logouturi="oams
so/logout.html")
```

11.3.5.3 Setting the Order of Providers

Perform these steps to set the order of the providers:

1. Log in to Weblogic 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 OAM Identity Asserter, OID/OVD Authenticator, and Default Authenticator by ensuring that the control flag for each authenticator is set as follows:
   - OAM Identity Asserter: REQUIRED
   - OID LDAP Authenticator (or OVD LDAP Authenticator): SUFFICIENT
   - Default Authenticator: SUFFICIENT
5. Click **OK**.

11.4 Validating Access Through Oracle HTTP Server and SSO

Validate single sign-on through both Oracle HTTP Server instances using the following URLs:

- http://WEBHOST1:7777/console
- http://WEBHOST1:7777/em
- http://WEBHOST1:7777/cs
- http://WEBHOST1:7777/imaging
- http://WEBHOST2:7777/console
- http://WEBHOST2:7777/em
- http://WEBHOST2:7777/cs
- http://WEBHOST2:7777/imaging

Then validate single sign-on through the front-end (using the SSO username and password):

- http://admin.mycompany.com/console
- http://admin.mycompany.com/em
- http://ecm.mycompany.com/cs
- http://ecm.mycompany.com/imaging

11.5 Backing Up the Installation

After you have verified that the extended domain is working, back up the installation. This is a quick backup for the express purpose of immediate restore in case of problems in the further steps. The backup destination is the local disk. This backup can be discarded once the enterprise deployment setup is complete. At 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 this guide. For information on how to recover components, see "Recovery of Components" and "Recovery After Loss of Component" sections in the guide. For recommendations specific to recovering from the loss of a host, see the "Recovering Oracle HTTP Server to a Different Host" in the guide. Also refer to the *Oracle Database Backup and Recovery User's Guide* for information on database backup.

Perform these steps 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. Perform a backup to save your domain configuration. The configuration files all exist under the *ORACLE_BASE*/admin/*domain_name* directory.

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

12 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:

- Section 12.1, "Monitoring the Topology"
- Section 12.2, "Defining an Optimal Input File Strategy for Oracle I/PM"
- Section 12.3, "Deploying Composites and Artifacts in SOA Enterprise Deployment Topology"
- Section 12.4, "Managing Space in the SOA Infrastructure Database"
- Section 12.5, "Configuring UMS Drivers"
- Section 12.6, "Scaling the Topology"
- Section 12.7, "Performing Backups and Recoveries"
- Section 12.8, "Troubleshooting"
- Section 12.9, "Best Practices"

12.1 Monitoring the Topology

For information on monitoring the Oracle ECM topology, see the following documents:

- *Oracle Fusion Middleware System Administrator's Guide for Content Server*
- *Oracle Fusion Middleware Application Administrator's Guide for Content Server*
- *Oracle Fusion Middleware Administrator's Guide for Imaging and Process Management*

12.2 Defining an Optimal Input File Strategy for Oracle I/PM

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 I/PM cluster:

- All of the machines in an Oracle I/PM cluster share a common input directory.
- Input files from this directory are distributed to each machine via 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 via the Work Manager created within the Oracle I/PM deployment.

Optimum performance will be achieved when:

- Each Oracle I/PM cluster instance has the maximum affordable number of parsing agents configured via the Work Manager without compromising the performance of the other I/PM 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.

12.3 Deploying Composites and Artifacts in SOA Enterprise Deployment Topology

When deploying SOA composites to the SOA subsystem used by I/PM, deploy to a specific server's address and not to the LBR address (ecm.mycompany.com). Deploying to the LBR address may require direct connection from the deployer nodes to the external LBR address, which may require additional ports to be opened in the firewalls used by the system.

12.4 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 the Oracle Enterprise Manager Fusion Middleware Control console (dashboard for instances). Search also in the SOA 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 the Oracle Enterprise Manager Fusion Middleware Control for this type of operation as in most cases the operations cause a transaction timeout. There are specific packages provided with the Repository Creation Utility to purge instances. For example:

```
DECLARE
  FILTER INSTANCE_FILTER := INSTANCE_FILTER();

   MAX_INSTANCES NUMBER;
  DELETED_INSTANCES NUMBER;
  PURGE_PARTITIONED_DATA BOOLEAN := TRUE;
 BEGIN
   .
  FILTER.COMPOSITE_PARTITION_NAME:='default';
  FILTER.COMPOSITE_NAME := 'FlatStructure';
  FILTER.COMPOSITE_REVISION := '10.0';
  FILTER.STATE := fabric. STATE_UNKNOWN;
  FILTER.MIN_CREATED_DATE := to_timestamp('2010-09-07','YYYY-MM-DD');
  FILTER.MAX_CREATED_DATE := to_timestamp('2010-09-08','YYYY-MM-DD');
  MAX_INSTANCES := 1000;
 .
  DELETED_INSTANCES := FABRIC.DELETE_COMPOSITE_INSTANCES(
    FILTER => FILTER,
    MAX_INSTANCES => MAX_INSTANCES,
    PURGE_PARTITIONED_DATA => PURGE_PARTITIONED_DATA
  );
```

This deletes the first 1,000 instances of the FlatStructure composite (version 10) created between '2010-09-07' and '2010-09-08' that are in "UNKNOWN" state. Refer to Chapter 8, "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.

12.5 Configuring UMS Drivers

Note: This step is required only if the SOA system used by Oracle I/PM is using Unified Messaging System (UMS).

UMS driver configuration is not automatically propagated in a SOA cluster. When UMS is used by the SOA system that I/PM 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").

In order to create the file in preparation for possible failovers, users can force a server migration and copy the file from the source node. For example, for I/PM:

1. Configure the driver for WLS_IPM1 in SOAHOST1.
2. Force a failover of WLS_IPM1 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. Do a remote copy of the driver configuration file from SOAHOST1 to SOAHOST2:

   ```
   SOAHOST1> 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). Perform these steps to restart the driver:

1. Log in to the Oracle WebLogic Administration console.
2. Expand the environment node on the navigation tree.
3. Click **Deployments**.
4. Select the driver.
5. Click **Stop->When work completes** and confirm the operation.
6. Wait for the driver to transition to the "Prepared" state (refresh the administration console page, if required).
7. Select the driver again, and click **Start->Servicing all requests** and confirm the operation.

Make sure that you verify in Oracle Enterprise Manager Fusion Middleware Control that the properties for the driver have been preserved.

12.6 Scaling the Topology

You can scale out and or scale up the enterprise topology. When you scale up the topology, you add new managed servers to nodes that are already running on one or more managed servers. When you scale out the topology, you add new managed servers to new nodes.

This section covers includes the topics:

- Section 12.6.1, "Scaling Up the Topology (Adding Managed Servers to Existing Nodes)"
- Section 12.6.2, "Scaling Out the Topology (Adding Managed Servers to New Nodes)"

Note: To scale out and up the SOA subsystem used by I/PM, refer to the SOA enterprise deployment topology documentation.

12.6.1 Scaling Up the Topology (Adding Managed Servers to Existing Nodes)

When scaling up the topology, 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 an Oracle Fusion Middleware home in shared storage. Use these existing installations (such as WebLogic Server home, Oracle Fusion 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`.

12.6.1.1 Scale-up Procedure for Oracle I/PM

Perform these steps to scale up the topology for Oracle I/PM:

1. Using the Oracle WebLogic Server Administration Console, clone WLS_IPM1 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.

   Perform these steps to clone a managed server:

   a. In the Domain Structure window of the Oracle WebLogic Server Administration Console, expand the **Environment** node and then **Servers**. The Summary of Servers page opens.

   b. Click **Lock & Edit** and then select the managed server that you want to clone (WLS_IPM1).

   c. Click **Clone**.

   d. Name the new managed server WLS_IPM*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 ECMHOST1, which is already running WLS_IPM1.

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 virtual host name for the server. This virtual host name should be different from the one used for the existing managed server.

   > **Note:** You must enable a VIP on ECMHOST1 (say, ECMHOST1VHN2), 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 6.2, "Enabling SOAHOST1VHN1 on SOAHOST1 and SOAHOST2VHN1 on SOAHOST2."

3. Configure a JMS persistence store and JMS servers for Oracle I/PM 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 I/PM are configured with no persistent store and use WebLogic Server's store (*ORACLE_BASE*/admin/*domain_name*/mserver/*domain_name*/servers/*server_name*/data/store/default). You must change Oracle I/PM's JMS server persistent store to use a shared base directory as follows:

   a. Log in to the Oracle WebLogic Server Administration Console.

b. In the Domain Structure window, expand the **Services** node and then click the **Persistence Stores** node. The Summary of Persistence Stores page opens.

c. Click **Lock & Edit**.

d. Click **New**, and then **Create FileStore**. The Create a New File Store page opens.

e. Enter the following information:

   - **Name:** IPMJMSServer*n*Store (for example, IPMJMSServer3Store, which allows you identify the service it is created for)

   - **Target:** WLS_IPM*n* (for example, WLS_IPM3).

   - **Directory:** Specify a directory that is located in shared storage so that it is accessible from both ECMHOST1 and ECMHOST2 (*ORACLE_BASE*/admin/*domain_name*/ipm_cluster/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**. The Summary of JMS Servers page opens.

h. Click **New** and then enter the following information:

   - **Name:** IpmJmsServer*n* (for example, IpmJmsServer3)

   - **Persistent Store:** select the persistence store that you created above: IPMJMSServer*n*Store (for example, IPMJMSServer3Store).

   Click **Next** and then specify 'WLS_IPM*n*' (for example, WLS_IPM3) as the target. Click **Finish** and activate the changes.

i. Click on the IpmJmsServer3 JMS server (represented as a hyperlink) in the Name column of the table. The settings page for the JMS server opens.

j. Click **Lock & Edit**.

k. In the Persistent Store drop-down list, select IPMJMSServer*n*Store.

l. Click **Save** and activate the changes.

4. Configure a default persistence store for WLS_IPM*n* for transaction recovery:

   a. Log in to the Oracle WebLogic Server Administration Console.

   b. In the Domain Structure window, expand the **Environment** node and then click the **Servers** node. The Summary of Servers page opens.

   c. Click WLS_IPM*n* (represented as a hyperlink) in the Name column of the table. The settings page for the WLS_IPM*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/ipm_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.

5. Disable host name verification for the new managed server. Before you can start and verify the WLS_IPM*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 Oracle WebLogic Administration Server and Node Manager in ECMHOST*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).

   Perform these steps to disable host name verification:

   a. In the Oracle Enterprise Manager Console, select Oracle WebLogic Server Administration Console, and log in.

   b. In the Domain Structure window, expand the **Environment** node and click **Servers**. The Summary of Servers page opens.

   c. Click WLS_IPM*n* (represented as a hyperlink) in the Name column of the table. The settings page for the WLS_IPM*n* server opens with the Configuration tab active.

   d. Open the SSL tab.

   e. Expand the **Advanced** section of the page.

   f. Click **Lock & Edit**.

   g. Set host name verification to 'None'.

   h. Click **Save**.

6. Start the newly created managed server (WLS_IPM):

   a. Log in to the Oracle WebLogic Server Administration Console.

   b. In the Domain Structure window, expand the **Environment** node and then click **Servers**. The Summary of Servers page opens.

   c. Open the Control tab, and shut down all existing WLS_IPM*n* managed servers in the cluster.

   d. Ensure that the newly created managed server, WLS_IPM*n*, is running.

7. 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 managed Oracle I/PM server should also be already present.

   Perform these steps to configure server migration:

   a. Log in to the Oracle WebLogic Server Administration Console.

   b. In the Domain Structure window, expand the **Environment** node and then click **Servers**. The Summary of Servers page opens.

c. 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. The settings page for the selected server opens.

d. 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 ECMHOST1, which is already running WLS_IPM1, select ECMHOST2. For new managed servers on ECMHOST2, which is already running WLS_IPM2, select ECMHOST1.

   > **Note:** The appropriate resources must be available to run the managed servers concurrently during migration.

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.

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_IPM*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_IPMn
     ```

   - Watch the Node Manager Console for a message indicating that WLS_IPM1's floating IP has been disabled.
   - Wait for Node Manager to attempt a second restart of WLS_IPM*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 Oracle 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.

12.6.1.2 Scale-up Procedure for Oracle UCM

Only one UCM managed server per node per domain is supported by Oracle Fusion Middleware. To add additional UCM managed servers, follow the steps in Section 12.6.2.2, "Scale-out Procedure for Oracle UCM" to add a UCM managed server to a new node.

12.6.1.3 Scale-up Procedure for SOA

Perform these steps for scaling up the SOA servers in the topology:

1. Using the Oracle 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.

   Perform these steps to clone a managed server:

   a. In the Domain Structure window of the Oracle WebLogic Server Administration Console, expand the **Environment** node and then **Servers**. The Summary of Servers page opens.

   b. 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 SOA and UMS on the new managed server:

   a. Use the Oracle 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 2.3, "Shared Storage and Recommended Directory Structure":

   ```
   ORACLE_BASE/admin/domain_name/cluster_name/jms/SOAJMSFileStore_N
   ```

   > **Note:** This directory must exist before the managed server is started or the start operation fails.

   b. Create a new JMS server for SOA (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 2.3, "Shared Storage and Recommended Directory Structure":

   ```
   ORACLE_BASE/admin/domain_name/cluster_name/jms/UMSJMSFileStore_N
   ```

   > **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 Oracle WebLogic Server Administration Console and then expand the **Messaging** node. Choose **JMS Modules** in the Domain Structure window. The JMS Modules page appears. Click **SOAJMSModule** (represented as a hyperlink in the Names column of the table). The Settings page for SOAJMSModule appears. Click 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 SOA 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 Oracle WebLogic Server Administration Console and then expand the **Messaging** node. Choose **JMS Modules** in the Domain Structure window. The JMS Modules page appears. Click **UMSJMSSystemResource** (represented as a hyperlink in the Names column of the table). The Settings page for UMSJMSSystemResource appears. Click 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 6.5, "Configuring Oracle Coherence for Deploying Composites."

Note: Only the localhost field needs to be changed for the server. Replace the localhost 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 2.3, "Shared Storage and Recommended Directory Structure").

   From the Administration Console, select the server name 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.

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 Oracle WebLogic 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 settings is propagated to the cloned server).

   Perform these steps to disable host name verification:

   a. In the Oracle Enterprise Manager Console, select Oracle WebLogic Server Administration Console, and log in.

   b. Expand the **Environment** node in the Domain Structure window.

   c. Click **Servers**. The Summary of Servers page opens.

   d. Select WLS_SOA*n* in the Names column of the table. The settings page for the server opens.

   e. 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**.

7. Start and test the new managed server from the Oracle WebLogic Server Administration Console:

   a. Ensure that the newly created managed server, WLS_SOA*n*, is running.

   b. Access the application on the LBR (https://ecm.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 Oracle HTTP Server's `mod_wl_ohs.conf` file. 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 managed SOA server should also be already present.

   Perform these steps to configure server migration:

   a. Log in to the Oracle WebLogic Server Administration Console.

   b. In the Domain Structure window, expand the **Environment** node and then click **Servers**. The Summary of Servers page opens.

c. 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. The settings page for the selected server opens.

d. 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. 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.

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 Oracle 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.

12.6.2 Scaling Out the Topology (Adding Managed Servers to New Nodes)

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 steps below.
- 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.

12.6.2.1 Scale-out Procedure for Oracle I/PM

Perform these steps to scale out the Oracle I/PM servers in the topology:

1. On the new node, mount the existing Middleware home, which should include the ECM 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:

   ```
   ECMHOSTn> cd ORACLE_COMMON_HOME/oui/bin/
   ECMHOSTn> ./attachHome.sh -jreLoc ORACLE_BASE/product/fmw/jrockit_160_<version>
   ```

   To update the Middleware home list, create (or edit, if another WebLogic installation exists in the node) the *MW_HOME*/bea/beahomelist file and add *ORACLE_BASE*/product/fmw to it.
3. Log in to the Oracle WebLogic Administration Console.
4. Create a new machine for the new node that will be used, and add the machine to the domain.
5. Update the machine's Node Manager address to map the IP of the node that is being used for scale-out.
6. Use the Oracle WebLogic Server Administration Console to clone WLS_IPM1 into a new managed server. Name it WLS_IPM*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 6.2, "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 I/PM on the new managed server:

   a. Use the Oracle WebLogic Server Administration Console to first create a new persistent store for the new IPMJMSServer (which will be created in a later step) and name it, for example, IPMJMSFileStore_N. Specify the path for the store as recommended in Section 2.3, "Shared Storage and Recommended Directory Structure" 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 I/PM; for example, IPMJMSServer_N. Use the IPMJMSFileStore_N created above for this JMS server. Target the IPMJMSServer_N server to the recently created managed server (WLS_IPM*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 is not required. Instead, the new nodes mount the already existing domain directory and use it for the new added managed server.

   ```
   SOAHOST1> cd ORACLE_COMMON_HOME/common/bin
   SOAHOST1> ./pack.sh -managed=true -domain=ORACLE_BASE/admin/domain_name/
   aserver/domain_name -template=edgdomaintemplateScale.jar -template_
   name=edgdomain_templateScale
   ```

10. Run the following command on SOAHOST1 to copy the created template file to ECMHOST*n*:

    > **Note:** If the new host, ECMHOST*n* or SOAHOST*n*, will use the same MW_HOME as SOAHOST1, then this step is not required.

    ```
    SOAHOST1> scp edgdomaintemplateScale.jar oracle@ECMHOSTn:/ORACLE_COMMON_HOME/
    common/bin
    ```

11. Run the `unpack` command on ECMHOST*n* to unpack the template in the managed server domain directory:

    ```
    ECMHOSTn > cd ORACLE_COMMON_HOME/common/bin
    ECMHOSTn > ./unpack.sh -domain=ORACLE_BASE/admin/domain_name/mserver/domain_
    name -template= edgdomaintemplateScale.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_IPM*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 Oracle WebLogic Administration Server and Node Manager in ECMHOST*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).

    Perform these steps to disable host name verification:

    a. In the Oracle Enterprise Manager Console, select Oracle WebLogic Server Administration Console.

    b. Expand the **Environment** node in the Domain Structure window.

    c. Click **Servers**. The Summary of Servers page opens.

    d. Select WLS_IPM*n* in the Names column of the table. The settings page for the server opens.

    e. 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**.

13. 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:

    ```
    ECMHOSTn > WL_HOME/server/bin/startNodeManager.sh New_Node_IP
    ```

    > **Note:** If you used the paths shown in Chapter 3.3.1, "Installing Oracle WebLogic Server and Creating the Fusion Middleware Home," *WL_HOME* would be *ORACLE_BASE*/product/fmw/wlserver_10.3.

14. Start and test the new managed server from the Oracle WebLogic Server Administration Console:

    a. Ensure that the newly created managed server, WLS_IPM*n*, is running.

    b. Access the application on the LBR (https://ecm.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 Oracle HTTP Server's `mod_wl_ohs.conf` file. 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.

15. 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. Verify the privileges defined in the new node to make sure server migration will work. Refer to Chapter 10, "Configuring Server Migration" for more details on privilege requirements.

Perform these steps to configure server migration:

a. Log in to the Oracle WebLogic Server Administration Console.

b. In the Domain Structure window, expand the **Environment** node and then click **Servers**. The Summary of Servers page opens.

c. Click the name of the server (represented as a hyperlink) in Name column of the table for which you want to configure migration. The settings page for the selected server opens.

d. 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.

16. 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_IPM*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_IPMn
```

- Watch the Node Manager Console for a message indicating that WLS_IPM1's floating IP has been disabled.
- Wait for Node Manager to attempt a second restart of WLS_IPM*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 Oracle 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.

12.6.2.2 Scale-out Procedure for Oracle UCM

Perform these steps to scale out the Oracle UCM servers in the topology:

Note: These steps assume that you are adding a new UCM 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 ECM 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:

   ```
   ECMHOSTn> cd ORACLE_COMMON_HOME/oui/bin/
   ECMHOSTn> ./attachHome.sh -jreLoc ORACLE_BASE/product/fmw/jrockit_160_<version>
   ```

   To update the Middleware home list, create (or edit, if another WebLogic installation exists in the node) the *MW_HOME*/bea/beahomelist file and add *ORACLE_BASE*/product/fmw to it.
3. Log in to the Oracle WebLogic Administration Console.
4. Create a new machine for the new node that will be used, and add the machine to the domain.
5. Update the machine's Node Manager address to map the IP of the node that is being used for scale-out.
6. Use the Oracle WebLogic Server Administration Console to clone WLS_UCM1 into a new managed server. Name it WLS_UCM*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 ECMHOST*n* to use for the new managed server as the listen address of the 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 new added managed server.

   ```
   SOAHOST1> cd ORACLE_COMMON_HOME/common/bin
   SOAHOST1> ./pack.sh -managed=true -domain=ORACLE_BASE/admin/domain_name/
   aserver/domain_name -template=edgdomaintemplateScale.jar -template_
   name=edgdomain_templateScale
   ```

9. Run the following command on SOAHOST1 to copy the created template file to ECMHOST*n*:

```
SOAHOST1> scp edgdomaintemplateScale.jar oracle@ECMHOSTn:/ORACLE_COMMON_HOME/
common/bin
```

10. Run the `unpack` command on ECMHOST*n* to unpack the template in the managed server domain directory:

```
ECMHOSTn > cd ORACLE_COMMON_HOME/common/bin
ECMHOSTn > ./unpack.sh -domain=ORACLE_BASE/admin/domain_name/mserver/domain_
name -template= edgdomaintemplateScale.jar -app_dir=ORACLE_BASE/admin/domain_
name/mserver/applications
```

11. 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:

```
ECMHOSTn> WL_HOME/server/bin/startNodeManager.sh New_Node_IP
```

> **Note:** If you used the paths shown in Chapter 3.3.1, "Installing Oracle WebLogic Server and Creating the Fusion Middleware Home," *WL_HOME* would be *ORACLE_BASE*/product/fmw/wlserver_10.3.

12. Disable host name verification for the new managed server. Before you can start and verify the WLS_UCM*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 Oracle WebLogic Administration Server and Node Manager in ECMHOST*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).

    Perform these steps to disable host name verification:

    a. In the Oracle Enterprise Manager Console, select Oracle WebLogic Server Administration Console.

    b. Expand the **Environment** node in the Domain Structure window.

    c. Click **Servers**. The Summary of Servers page opens.

    d. Select WLS_UCM*n* in the Names column of the table. The settings page for the server opens.

    e. 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**.

13. Start and test the new managed server from the Oracle WebLogic Server Administration Console:

    a. Ensure that the newly created managed server, WLS_UCM*n*, is running.

    b. Access the application on the LBR (https://ecm.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 Oracle HTTP Server's `mod_wl_ohs.conf` file. 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.

12.6.2.3 Scale-out Procedure for SOA

Perform these steps to scale out the SOA servers in the topology:

1. On the new node, mount the existing Middleware home, which should include the SOA 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:

   ```
   SOAHOSTn> cd ORACLE_COMMON_HOME/oui/bin/
   SOAHOSTn> ./attachHome.sh -jreLoc ORACLE_BASE/product/fmw/jrockit_160_<version>
   ```

   To update the Middleware home list, create (or edit, if another WebLogic installation exists in the node) the *MW_HOME*/bea/beahomelist file and add *ORACLE_BASE*/product/fmw to it.
3. Log in to the Oracle WebLogic Administration Console.
4. Create a new machine for the new node that will be used, and add the machine to the domain.
5. Update the machine's Node Manager address to map the IP of the node that is being used for scale-out.
6. Use the Oracle 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 new added managed server.

```
SOAHOST1> cd ORACLE_COMMON_HOME/common/bin
SOAHOST1> ./pack.sh -managed=true -domain=ORACLE_BASE/admin/domain_name/
aserver/domain_name -template=edgdomaintemplateScale.jar -template_
name=edgdomain_templateScale
```

9. Run the following command on SOAHOST1 to copy the created template file to SOAHOST*n*:

```
SOAHOST1> scp edgdomaintemplateScale.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:

```
SOAHOSTn > cd ORACLE_COMMON_HOME/common/bin
SOAHOSTn > ./unpack.sh -domain=ORACLE_BASE/admin/domain_name/mserver/domain_
name -template= edgdomaintemplateScale.jar -app_dir=ORACLE_BASE/admin/domain_
name/mserver/applications
```

11. Create JMS servers for SOA and UMS on the new managed server:

    a. Use the Oracle 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 2.3, "Shared Storage and Recommended Directory Structure":

    ```
    ORACLE_BASE/admin/domain_name/cluster_name/jms/SOAJMSFileStore_N
    ```

    Note: This directory must exist before the managed server is started or the start operation fails.

    b. Create a new JMS server for SOA (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 2.3, "Shared Storage and Recommended Directory Structure":

    ```
    ORACLE_BASE/admin/domain_name/cluster_name/jms/UMSJMSFileStore_N
    ```

    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 Oracle WebLogic Server Administration Console and then expand the **Messaging** node. Choose **JMS Modules** in the Domain Structure window. The JMS Modules page appears. Click **SOAJMSModule** (represented as a hyperlink in the Names column of the table). The Settings page for SOAJMSModule appears. Click 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 SOA 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 Oracle WebLogic Server Administration Console and then expand the **Messaging** node. Choose **JMS Modules** in the Domain Structure window. The JMS Modules page appears. Click **UMSJMSSystemResource** (represented as a hyperlink in the Names column of the table). The Settings page for UMSJMSSystemResource appears. Click 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 2.3, "Shared Storage and Recommended Directory Structure").

From the Administration Console, select the server name 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.

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 Oracle WebLogic 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).

Perform these steps to disable host name verification:

a. In the Oracle Enterprise Manager Console, select Oracle WebLogic Server Administration Console.

b. Expand the **Environment** node in the Domain Structure window.

c. Click **Servers**. The Summary of Servers page opens.

d. Select WLS_SOA*n* in the Names column of the table. The settings page for the server opens.

e. 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:

```
SOAHOSTn> WL_HOME/server/bin/startNodeManager.sh New_Node_IP
```

> **Note:** If you used the paths shown in Chapter 3.3.1, "Installing Oracle WebLogic Server and Creating the Fusion Middleware Home," *WL_HOME* would be *ORACLE_BASE*/product/fmw/wlserver_10.3.

15. Start and test the new managed server from the Oracle WebLogic Server Administration Console:

    a. Ensure that the newly created managed server, WLS_SOA*n*, is running.

    b. Access the application on the LBR (https://ecm.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 Oracle HTTP Server's `mod_wl_ohs.conf` file. 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. 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 managed SOA server is already present in the new node.

Perform these steps to configure server migration:

a. Log in to the Oracle WebLogic Server Administration Console.

b. In the Domain Structure window, expand the **Environment** node and then click **Servers**. The Summary of Servers page opens.

c. Click the name of the server (represented as a hyperlink) in Name column of the table for which you want to configure migration. The settings page for the selected server opens.

d. 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.

17. 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 Oracle 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.

12.7 Performing Backups and Recoveries

Table 12–1 lists the static artifacts to back up in the 11*g* Oracle ECM enterprise deployment.

Table 12–1 Static Artifacts to Back Up in the 11g ECM Enterprise Deployment

Type	Host	Location	Tier
ORACLE HOME (DB)	CUSTDBHOST1 and CUSTDBHOST	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 home as well)	SOAHOST1 and SOAHOST2*	*MW_HOME* The SOA home is also under *MW_HOME*: *ORACLE_HOME*	Application tier

Table 12–1 (Cont.) Static Artifacts to Back Up in the 11g ECM Enterprise Deployment

Type	Host	Location	Tier
Installation-related files		OraInventory, *User_Home*/bea/beahomelist, oraInst.loc, oratab	N/A

* ECMHOST1 and ECMHOST2 use the binaries installed from SOAHOST1 and SOAHOST2. Backup is centralized in SOAHOST1 and SOAHOST2.

Table 12–2 lists the run-time artifacts to back up in the 11*g* ECM enterprise deployment.

Table 12–2 Run-Time Artifacts to Back Up in the 11g ECM Enterprise Deployment

Type	Host	Location	Tier
Application artifacts (EAR and WAR files)	SOAHOST1, SOAHOST2, ECMHOST1, and ECMHOST2	Find the application artifacts by viewing all of the deployments through the administration console.	Application tier
SOA runtime artifacts	SOAHOST1 or SOAHOST2	*ORACLE_BASE*/admin/*domain_name*/*soa_cluster_name*	Application tier
UCM runtime artifacts	ECMHOST1 or ECMHOST2	*ORACLE_BASE*/admin/*domain_name*/*ucm_cluster_name*	Application tier
I/PM runtime artifacts	ECMHOST1 or ECMHOST2	*ORACLE_BASE*/admin/*domain_name*/*ipm_cluster_name*	Application tier
Customized managed server configuration for ECM	ECMHOST1 or ECMHOST2	*ORACLE_BASE*/admin/*domain_name*/mserver/*domain_name*/ucm/cs/bin/intradoc.cfg and *ORACLE_BASE*/admin/*domain_name*/mserver/*domain_name*/server_migration/wlsifconfig.sh	Application tier
Customized managed server configuration for SOA	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 WLS during deployment, for example, "3682yq"). and *ORACLE_BASE*/admin/*domain_name*/mserver/*domain_name*/server_migration/wlsifconfig.sh	Application tier
OHS 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*.

12.8 Troubleshooting

This section covers the following topics:

- Section 12.8.1, "Page Not Found When Accessing soa-infra Application Through Load Balancer"
- Section 12.8.2, "Soa-infra Application Fails to Start Due to Deployment Framework Issues (Coherence)"
- Section 12.8.3, "Incomplete Policy Migration After Failed Restart of SOA Server"
- Section 12.8.4, "SOA, I/PM, or UCM Servers Fail to Start Due to Maximum Number of Processes Available in Database"
- Section 12.8.5, "Administration Server Fails to Start After a Manual Failover"
- Section 12.8.6, "Error While Activating Changes in Administration Console"
- Section 12.8.7, "SOA or I/PM Server Not Failed Over After Server Migration"
- Section 12.8.8, "SOA or I/PM Server Not Reachable From Browser After Server Migration"
- Section 12.8.9, "OAM Configuration Tool Does Not Remove URLs"
- Section 12.8.10, "Redirecting of Users to Login Screen After Activating Changes in Administration Console"
- Section 12.8.11, "Redirecting of Users to Administration Console's Home Page After Activating Changes to OAM"
- Section 12.8.12, "Configured JOC Port Already in Use"
- Section 12.8.13, "Using CredentialAccessPermissions to Allow Oracle I/PM to Read Credentials From the Credential Store"
- Section 12.8.14, "Improving Performance with Very Intensive Document Uploads from Oracle I/PM to Oracle UCM"
- Section 12.8.15, "Out-of-Memory Issues on Managed Servers"
- Section 12.8.16, "Regenerating the Master Password for Oracle UCM Servers"
- Section 12.8.17, "Logging Out From the WebLogic Server Administration Console Does Not End the User Session"

12.8.1 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 SOA servers appear as "Running" in the WLS Administration Console.

Solution: Even when the SOA managed servers may be up and running, some of the applications contained in them may be in Admin, Prepared or other states different from Active. The soa-infra application may be unavailable while the SOA server is running. Check the Deployments page in the Administration Console to verify the status of the soa-infra application. It should be in "Active" state. Check the SOA 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.

12.8.2 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 SOA 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, a message similar to the above may appear if a multicast conflict arises when starting the soa-infra application (that is, starting the managed server on which SOA 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 *Oracle Coherence Developer's Guide for Oracle Coherence*.

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
   ```

12.8.3 Incomplete Policy Migration After Failed Restart of SOA Server

Problem: The SOA 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 SOA 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 SOA 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>
```

12.8.4 SOA, I/PM, or UCM Servers Fail to Start Due to Maximum Number of Processes Available in Database

Problem: A SOA, I/PM, or UCM server fails to start. The domain has been extended for new types of managed server (for example, UCM extended for I/PM) or the system has been scaled up (added new servers of the same type). The SOA, I/PM, or UCM 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:

```
SQL> SHOW PARAMETER processes
```

Set the initialization parameter using the following command:

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

Restart the database.

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

12.8.5 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.

12.8.6 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 clicking "Activate Changes":

```
An error occurred during activation of changes, please see the log for details.
 [Management:141190]The commit phase of the configuration update failed with an
exception:
In production mode, it's not allowed to set a clear text value to the property:
PasswordEncrypted of ServerStartMBean
```

Solution: This may happen when start parameters are changed for a server in the Administration Console. In this case, provide username/password information in the server start configuration in the Administration Console for the specific server whose configuration was being changed.

12.8.7 SOA or I/PM 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 SOA or I/PM 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.

12.8.8 SOA or I/PM Server Not Reachable From Browser After Server Migration

Problem: Server migration is working (SOA or I/PM 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 SOA or I/PM 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.

12.8.9 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 OAM. Log on to the Access Administration site for OAM, click on My Policy Domains, click on the created policy domain (SOA_EDG), then on the Resources tab, and remove the incorrect URLs.

12.8.10 Redirecting of Users to Login Screen After Activating Changes in Administration Console

Problem: After configuring OHS and LBR to access the Oracle WebLogic Administration Console, some activation changes cause the redirection to the login screen for the 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 `ecm.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.

12.8.11 Redirecting of Users to Administration Console's Home Page After Activating Changes to OAM

Problem: After configuring OAM, some activation changes cause the redirection to the Administration Console's home page (instead of the context menu where the activation was performed).

Solution: This is expected when OAM SSO is configured and 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" check box.

12.8.12 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.

12.8.13 Using CredentialAccessPermissions to Allow Oracle I/PM to Read Credentials From the Credential Store

Problem: Oracle I/PM 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 I/PM 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 I/PM 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 ECM, connect to the Administration Server, and execute the `grantIPMCredAccess()` command:

```
ECMHOST1> cd ORACLE_HOME/common/bin
ECMHOST1> ./wlst.sh
wls:/offline> connect()
wls:/ecmedg_domain/serverConfig> grantIPMCredAccess()
```

> **Note:** When connecting, provide the credentials and address for the Administration Server.

12.8.14 Improving Performance with Very Intensive Document Uploads from Oracle I/PM to Oracle UCM

Problem: If a host name-based security filter is used in Oracle UCM (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 I/PM to Oracle UCM. This is caused by the reverse DNS lookup which is required in Oracle UCM to allow the connections from the Oracle I/PM 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.

Perform these steps to change the host name-based security filter in Oracle UCM to an IP-based filter:

1. Open the file *ORACLE_BASE*/admin/*domain_name*/ucm_cluster/cs/config/config.cfg in a text editor.

2. Remove or comment out the following two lines:

   ```
   SocketHostNameSecurityFilter=localhost|localhost.mycompany.com|ecmhost1vhn1|
   ecmhost2vhn1
   AlwaysReverseLookupForHost=Yes
   ```

3. Add the IP addresses (listen addresses) of the WLS_IPM1 and WLS_IPM2 managed servers (ECMHOST1VHN1 and ECMHOST2VHN1, 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_IPM1 and WLS_IPM2, 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 UCM servers for the changes to take effect.

12.8.15 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 Oracle WebLogic Administration Console.
2. Click **Environment**, then **Servers**.
3. Click on 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 Fusion Middleware Installation Guide for Oracle Enterprise Content Management Suite* for further details.

7. Save the configuration changes.
8. Restart all running managed servers.

12.8.16 Regenerating the Master Password for Oracle UCM Servers

In case the cwallet.sso file of the Oracle UCM 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 UCM managed servers.
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/ucm_cluster/cs/config/private.
4. Start the WLS_UCM1 server in ECMHOST1.
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/ucm_cluster/cs/config/private.
6. Use Oracle UCM's System Properties command-line tool to update the passwords for the database.

7. Verify that the standalone Oracle UCM 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 UCM 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 ECMHOST2 and that the file is the same as created or updated by the Oracle UCM server in ECMHOST1.
10. Verify that the standalone Oracle UCM applications (Batchloader, System Properties, and so on) are working correctly.
11. Verify that the standalone Oracle UCM applications work correctly on both nodes at the same time.

12.8.17 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 OAM 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.

12.9 Best Practices

This section covers the following topics:

- Section 12.9.1, "Preventing Timeouts for SQLNet Connections"
- Section 12.9.2, "Auditing"
- Section 12.9.3, "Configuring Oracle Web Service Manager Security Policies for Oracle IP/M and Oracle UCM Web Services"

12.9.1 Preventing Timeouts for SQLNet Connections

Much of the EDG production deployment involves firewalls. Because database connections are made across firewalls, Oracle recommends that the firewall be configured so that the database connection is not timed out. For Oracle Real Application Clusters (RAC), the database connections are made on Oracle RAC VIPs and the database listener port. You must configure the firewall to not time out such connections. If such a configuration is not possible, set the`*SQLNET.EXPIRE_TIME=n*` parameter in the *ORACLE_HOME*/network/admin/sqlnet.ora file on the database server, where *n* is the time in minutes. Set this value to less than the known value of the timeout for the network device (that is, a firewall). For Oracle RAC, set this parameter in all of the Oracle home directories.

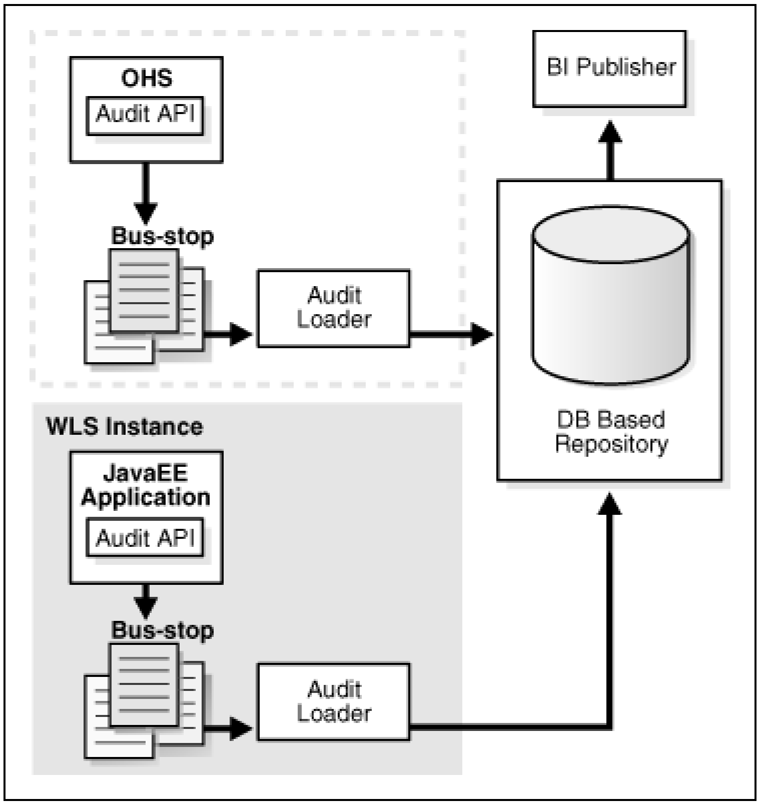
12.9.2 Auditing

Oracle Fusion Middleware Audit Framework is a new service in Oracle Fusion Middleware 11*g*, designed to provide a centralized audit framework for the middleware family of products. The framework provides audit service for platform components such as Oracle Platform Security Services (OPSS) and Oracle Web

Services. It also provides a framework for JavaEE applications, starting with Oracle's own JavaEE components. JavaEE applications will be able to create application-specific audit events. For non-JavaEE Oracle components in the middleware, such as C or JavaSE components, the audit framework also provides an end-to-end structure similar to that for JavaEE applications.

Figure 12–1 is a high-level architectural diagram of the Oracle Fusion Middleware Audit Framework.

Figure 12–1 Audit Event Flow



The Oracle Fusion Middleware Audit Framework consists of the following key components:

- **Audit APIs:** These are APIs provided by the audit framework for any audit-aware components integrating with the Oracle Fusion Middleware Audit Framework. During run time, applications may call these APIs, where appropriate, to audit the necessary information about a particular event happening in the application code. The interface allows applications to specify event details such as username and other attributes needed to provide the context of the event being audited.
- **Audit Events and Configuration:** The Oracle Fusion Middleware Audit Framework provides a set of generic events for convenient mapping to application audit events. Some of these include common events such as authentication. The framework also allows applications to define application-specific events.

  These event definitions and configurations are implemented as part of the audit service in Oracle Platform Security Services. Configurations can be updated through Enterprise Manager (UI) and WLST (command-line tool).

- **Audit Bus-stop:** Bus-stops are local files containing audit data before they are pushed to the audit repository. In the event where no database repository is configured, these bus-stop files can be used as a file-based audit repository. The bus-stop files are simple text files that can be queried easily to look up specific audit events. When a DB-based repository is in place, the bus-stop acts as an intermediary between the component and the audit repository. The local files are periodically uploaded to the audit repository based on a configurable time interval.
- **Audit Loader:** As the name implies, the audit loader loads the files from the audit bus-stop into the audit repository. In the case of platform and JavaEE application audit, the audit loader is started as part of the JavaEE container start-up. In the case of system components, the audit loader is a periodically spawned process.
- **Audit Repository:** The audit repository contains a predefined Oracle Fusion Middleware Audit Framework schema, created by Repository Creation Utility (RCU). Once configured, all the audit loaders are aware of the repository and upload data to it periodically. The audit data in the audit repository is expected to be cumulative and will grow over time. Ideally, this should not be an operational database used by any other applications; rather, it should be a standalone RDBMS used for audit purposes only. In a highly available configuration, Oracle recommends that you use an Oracle Real Application Clusters (RAC) database as the audit data store.
- **Oracle Business Intelligence Publisher:** The data in the audit repository is exposed through predefined reports in Oracle Business Intelligence Publisher. The reports allow users to drill down the audit data based on various criteria. For example:
  - Username
  - Time range
  - Application type
  - Execution context identifier (ECID)

For more introductory information for the Oracle Fusion Middleware Audit Framework, see the "Introduction to Oracle Fusion Middleware Audit Framework" chapter in the *Oracle Fusion Middleware Security Guide*.

For information on how to configure the repository for Oracle Fusion Middleware Audit Framework, see the "Configuring and Managing Auditing" chapter in the *Oracle Fusion Middleware Security Guide*.

The EDG topology does not include Oracle Fusion Middleware Audit Framework configuration. The ability to generate audit data to the bus-stop files and the configuration of the audit loader will be available once the products are installed. The main consideration is the audit database repository where the audit data is stored. Because of the volume and the historical nature of the audit data, it is strongly recommended that customers use a separate database from the operational store or stores being used for other middleware components.

12.9.3 Configuring Oracle Web Service Manager Security Policies for Oracle IP/M and Oracle UCM Web Services

When first installed, the Oracle I/PM and Oracle UCM web services are configured with no Oracle Web Service Manager (WSM) 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 IP/M and Oracle UCM web services, follow the steps in the *Oracle Fusion Middleware Developer's Guide for Imaging and Process Management* and the *Oracle Fusion Middleware Installation Guide for Oracle Enterprise Content Management Suite*.

Index