Oracle® Fusion Middleware

Enterprise Deployment Guide for Oracle SOA Suite

11*g* Release 1 (11.1.1)

E12036-07

April 2011

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

E12036-07

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

Primary Author: Joe Paul

Contributing Author: Fermin Castro Alonso

Contributor: Janga Aliminati, Pradeep Bhat, Kevin Cluggage, Richard Delval, Jordan Douglas, Eileen He, Rahul Menezes, Ranjit Mulye, Bharath Reddy, Michael Rhys, Praveen Sampath, Dhaval Shah, Sunita Sharma, Herb Stiel, Jianliang Yi, Zhiming Zhu

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

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

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

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

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

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

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

Contents

3 Installing Oracle HTTP Server

4 Creating a Domain

5 Extending the Domain for SOA Components

6 Extending the Domain to Include Oracle BPM

7 Extending the Domain to Include BAM

8 Setting Up Node Manager

9 Server Migration

10 Managing the Topology

11 Integration With Oracle Identity Management

12 Miscellaneous

Preface

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

Audience

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

Documentation Accessibility

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

Accessibility of Code Examples in Documentation

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

Accessibility of Links to External Web Sites in Documentation

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

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

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

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

Conventions

The following text conventions are used in this document:

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

1 Enterprise Deployment Overview

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

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 many Oracle products across the entire technology stack: Oracle Database, Oracle Fusion Middleware, and Enterprise Manager Fusion Middleware Control.

An Oracle Fusion Middleware enterprise deployment:

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

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

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

1.2 Terminology

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

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

node in a hardware cluster is a standalone server that runs its own processes. These processes can communicate with one another to form what looks like a single system that cooperatively provides applications, system resources, and data to users.

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

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

  Managed Server homes can also be optionally located in the shared disk. The shared storage can be a Network Attached Storage (NAS), a Storage Area Network (SAN) or any other storage system that multiple nodes can access simultaneously and can read-write.
- **primary node**: The node that is actively running an Oracle Fusion Middleware instance at any given time and has been configured to have a backup/secondary node. If the primary node fails, Oracle Fusion Middleware instance is failed over to the secondary node. This failover can be manual or automated using the Clusterware for Administration Server. For a server migration based scenario, WebLogic Whole Server Migration is used for automated failover.
- **secondary node**: The node that is the backup node for an Oracle Fusion Middleware instance. This is where the active instance fails over when the primary node is no longer available. See the definition for primary node in this section.
- **network host name**: Network host name is a name assigned to an IP address either through the `/etc/hosts` file or through DNS resolution. This name is visible in the network that the machine to which it refers to is connected. Often, the network host name and physical host name are identical. However, each machine has only one physical host name but may have multiple network host

names. Thus, a machine’s network host name may not always be its physical host name.

- **physical host name**: This guide differentiates between the terms physical host name and network host name. This guide uses physical host name to refer to the "internal name" of the current machine. On UNIX, this is the name returned by the `hostname` command.

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

- **physical IP**: Physical IP refers to the IP of a machine on the network. In almost all cases, it is normally associated with the physical host name of the machine (see the definition of the physical host name). In contrast to a virtual IP, it is always associated with the same machine when on a network.

- **switchover**: During normal operation, active members of a system may require maintenance or upgrading. A switchover process can be initiated to allow a substitute member to take over the workload performed by the member that requires maintenance or upgrading, which undergoes planned downtime. The switchover operation ensures continued service to consumers of the system.

- **switchback**: When a switchover operation is performed, a member of the system is deactivated for maintenance or upgrading. When the maintenance or upgrading is completed, the system can undergo a switchback operation to activate the upgraded member and bring the system back to the pre-switchover configuration.

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

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

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

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

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

1.3 Benefits of Oracle Recommendations

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

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

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

1.3.1 Built-in Security

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

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

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

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

1.3.2 High Availability

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

1.4 Hardware Requirements

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

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

Table 1–1 Typical Hardware Requirements

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

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

[2] BAM can reuse Middleware 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.

1.5 Enterprise Deployment Reference Topology

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

This guide provides configuration instructions for a reference enterprise topology that uses service-oriented architecture (SOA) with Oracle Access Manager, as shown in Figure 1–1, with Oracle Access Manager and Oracle Business Activity Monitoring (BAM), as shown in Figure 1–2, or with Oracle Access Manager and BPM, as shown in Figure 1–3.

Figure 1–1 MySOACompany Topology with Oracle Access Manager

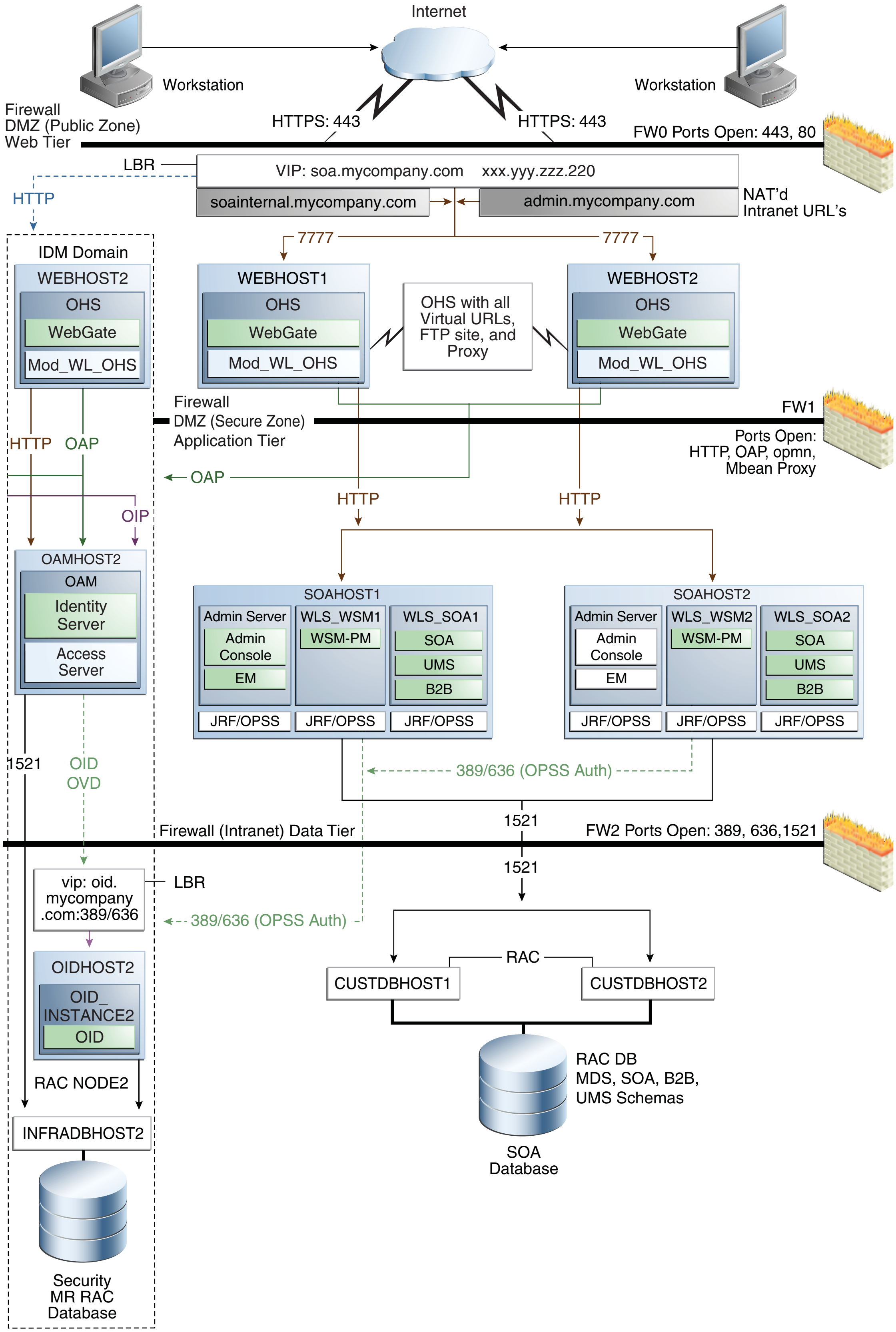


Figure 1–2 MySOACompany Topology with Oracle Access Manager and Business Activity Monitoring

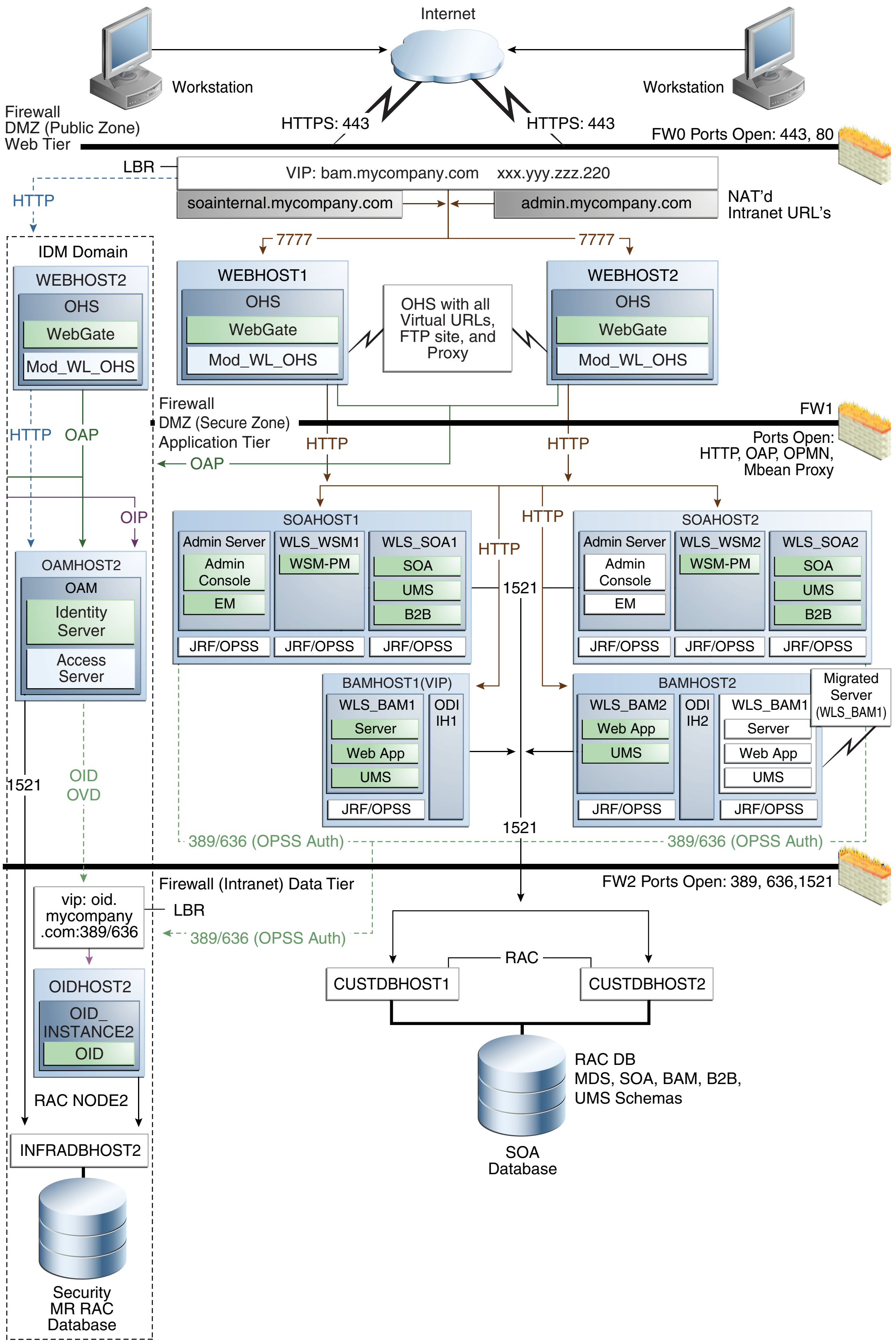
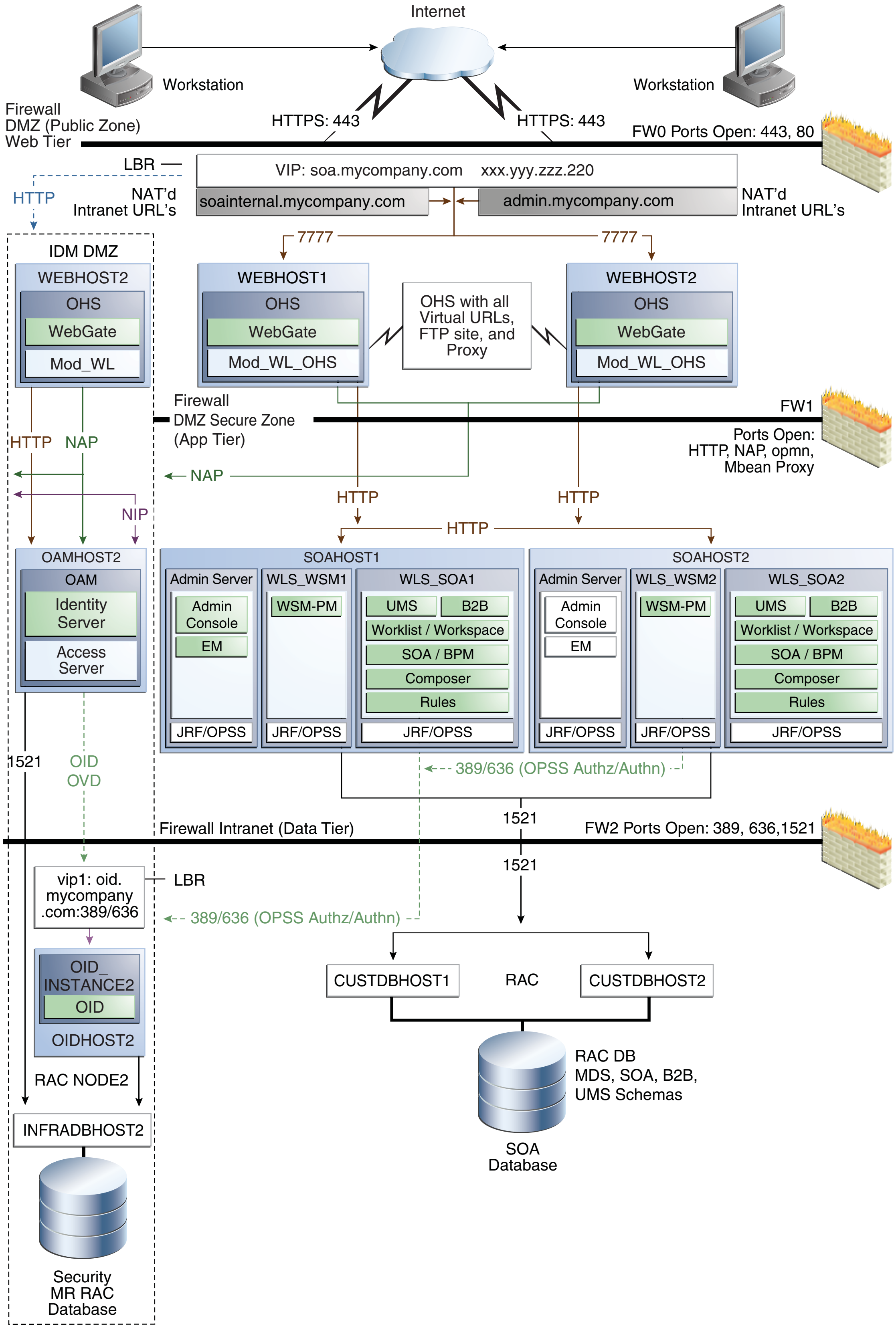


Figure 1–3 MySOACompany Topology with Oracle Access Manager and BPM



This section covers these topics:

1.5.1 Oracle Identity Management

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

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

1.5.2 Web Tier

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

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

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

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

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

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

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

1.5.2.1 Load Balancer Requirements

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

- Ability to load-balance traffic to a pool of real servers through a virtual host name: Clients access services using the virtual host name (instead of using actual host names). The load balancer can then load balance requests to the servers in the pool.

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

1.5.3 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 and B2B. The managed servers are configured in an active-active manner.

BAMHOST1 and BAMHOST2 run the BAM Server and BAM Web Applications.

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 4.22, "Manually Failing Over the Administration Server to SOAHOST2"); alternatively you can configure the Oracle WebLogic Server Administration Console with CFC/CRS to fail over automatically on a separate hardware cluster (not shown in this architecture).

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

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

1.5.4 Data Tier

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

In this tier, an Oracle RAC database runs on the nodes CUSTDBHOST1 and CUSTDBHOST2. The database contains the schemas needed by the SOA and BAM components. The BAM 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 Enterprise Deployment.

1.5.5 What to Install

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

Table 1–2 Components and Installation Sources

Component	Distribution Medium
Oracle Database 10*g* or 11*g*	Oracle Database CD (in 10*g* series, 10.2.0.4 or higher; in 11*g* series, 11.1.0.7 or higher)
Repository Creation Utility (RCU)	Oracle Fusion Middleware Repository Creation Utility 11*g* (11.1.1.4.0) DVD
Oracle WebLogic Server (WLS)	Oracle Weblogic Server 11g R1 (10.3.4) DVD
Oracle HTTP Server	Oracle Fusion Middleware WebTier and Utilities 11*g* (11.1.1.4.0) DVD
Oracle SOA Suite	Oracle SOA Suite 11g (11.1.1.4.0) DVD
Oracle Business Activity Monitor (BAM)	Oracle Fusion Middleware 11*g* (11.1.1.4.0) DVD
Oracle Access Manager 10*g* Webgate	Oracle Access Manager 10*g* Webgates (10.1.4.3.0) DVD ; OAM OHS 11*g* webgates per platform
Oracle Virtual Directory (OVD)	Oracle Identity Management 11*g* (11.1.1.4.0) DVD

1.5.6 Unicast Requirement

Oracle recommends that the nodes in the mySOACompany 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 muticast-enabled network to access multicast topics.)

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

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

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

1.6 How to Use This Guide

This section covers the following topics:

- Section 1.6.1, "Installation and Configuration Procedure"
- Section 1.6.2, "Overview of Installation Strategies"

1.6.1 Installation and Configuration Procedure

Table 1–3 summarizes the process by which you install and configure SOA Enterprise Deployment. Follow the procedures indicated in the first column, in the order shown, for the chosen configuration.

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

Table 1–3 SOA Installation Procedures

Perform the steps in...	To configure a domain with only Admin Server and WSM-PM	To configure a domain with Admin Server, WSM-PM, and extend it with a SOA cluster	To Configure a domain with Admin Server, WSM-PM, and extend it with a SOA/BPM cluster	To Configure a domain with Admin Server, WSM-PM, SOA and extend it to a BPM cluster	To configure a domain with Admin Server, WSM-PM, and BAM cluster (without SOA)	To configure a domain with Admin Server, WSM-PM, SOA/BPM cluster, and BAM cluster
Chapter 2, "Database and Environment Preconfiguration"	Yes	Yes	Yes	Yes	Yes	Yes
Chapter 3, "Installing Oracle HTTP Server"	Yes	Yes	Yes	Yes	Yes	Yes
Chapter 4, "Creating a Domain"	Yes	Yes	Yes	Yes	Yes	Yes
Chapter 5, "Extending the Domain for SOA Components"	No	Yes	No	Yes	No	Yes
Chapter 6, "Extending the Domain to Include Oracle BPM"	No	No	Yes (Use option 1)	Yes (Use option 2)	No	Yes (if BPM is used with BAM)
Chapter 7, "Extending the Domain to Include BAM"	No	No	No	No)	Yes)	Yes
Chapter 8, "Setting Up Node Manager"	Recommended (optional depending on the type of security required for the application tier)	Recommended (optional depending on the type of security required for the application tier)	Recommended (optional depending on the type of security required for the application tier)	Recommended (optional depending on the type of security required for the application tier)	Recommended (optional depending on the type of security required for the application tier)	Recommended (optional depending on the type of security required for the application tier)
Chapter 9, "Server Migration"	No	Yes (for production environments)	Yes (for production environments)	Yes (for production environments)	Yes	Yes

1.6.2 Overview of Installation Strategies

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

1. Run a first pass of the Configuration Wizard to install the Administration Server, Enterprise Manager, and WSM-PM (described in Chapter 4, "Creating a Domain").
2. Run a second pass of the Configuration Wizard to install the SOA components (described in Chapter 5, "Extending the Domain for SOA Components").
3. Optionally, run a third pass to install the BAM components (described in Chapter 7, "Extending the Domain to Include BAM").

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

Some variation from the above topology is possible. For example, if a deployment chooses to install BAM alone, then only sections applicable extend with BAM need to be followed. Also, in this case, it is expected that the Adminserver will exist on BAMHOST1 instead and the instructions on creating the domain should be modified appropriately.

Additionally, Oracle Fusion Middleware 11*g* R1 11.1.1.3 (PS2) introduces Oracle BPM as a superset of the service engines that existed in previous Oracle Fusion Middleware 11g SOA Suite releases. You can enable Oracle BPM Suite in their systems in different ways. Using the Configuration Wizard to extend an exiting SOA domain, or using the Configuration Wizard to extend an Admin+WSM domain with both SOA and BPM components. Both options are explained in Chapter 6, "Extending the Domain to Include Oracle BPM."

2 Database and Environment Preconfiguration

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

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

2.1 Database

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

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

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

This section covers the following topics:

- Section 2.1.1, "Setting Up the Database"
- Section 2.1.2, "Loading the Oracle Fusion Metadata Repository in the Oracle RAC Database"
- Section 2.1.4, "Backing Up the Database"

2.1.1 Setting Up the Database

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

- Section 2.1.1.1, "Database Host Requirements"

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

2.1.1.1 Database Host Requirements

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

- **Oracle Clusterware**

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

- **Automatic Storage Management** (optional)

  ASM gets installed for the node as a whole. It is recommended that you install it in a separate Oracle Home from the Database Oracle Home. This option comes in at runInstaller. In the Select Configuration page, select the Configure Automatic Storage Management option to create a separate ASM home.

2.1.1.2 Supported Database Versions

Oracle SOA 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 "Certified Databases" section in the Certification Document:

  `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 SOA Suite requires the database be 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 Assistant.

Table 2–1 Required Initialization Parameters

Configuration	Parameter	Required Value	Parameter Class
SOA	`PROCESSES`	300 or greater	Static
BAM	`PROCESSES`	100 or greater	Static

Table 2–1 (Cont.) Required Initialization Parameters

Configuration	Parameter	Required Value	Parameter Class
SOA and BAM	`PROCESSES`	400 or greater	Static

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

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

```
SQL> SHOW PARAMETER processes
```

Set the initialization parameter using the following command:

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

Restart the database.

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

2.1.1.4 Database Services

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

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

1. Use the `CREATE_SERVICE` subprogram to create the `soaedg.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 => 'soaedg.mycompany.com',
   NETWORK_NAME => 'soaedg.mycompany.com',
   );
   ```

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

   ```
   prompt> srvctl add service -d soadb -s soaedg -r soadb1,soadb2
   ```

3. Start the service using `srvctl`:

   ```
   prompt> srvctl start service -d soadb -s soaedg
   ```

> **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 *soadb.mycompany.com* and the default service is one with the same name. The SOA install is configured to use the service *soaedg.mycompany.com*. It is recommended that a service named *bamedg.mycompany.com* is used for BAM.

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

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

1. Start Repository Creation Utility (RCU), which is available from the RCU DVD or from the location listed in Table 1–2, by first inserting the RCU DVD.

   > **Note:** The RCU used to seed the database must match the patch set level of the SOA Suite installation. For example, if you install SOA Suite PS1 Enterprise Deployment you use RCU PS1. If you install SOA Suite PS2 Enterprise Deployment, you must use PS2 RCU.

2. Start RCU from the *bin* directory:

   ./rcu

3. In the Welcome screen, click **Next**.
4. In the Create Repository screen, select **Create** to load component schemas into a database. Click **Next**.
5. In the Database Connection Details screen, enter connect information for your database:

   - **Database Type**: select **Oracle Database**.
   - **Host Name**: Enter the name of the node that is running the database. For the Oracle RAC database, specify the VIP name or one of the node names as the host name: `CUSTDBHOST1-VIP`.
   - **Port**: Enter the port number for the database: 1521.
   - **Service Name**: Enter the service name of the database: soaedg.mycompany.com
   - **Username**: `SYS`
   - **Password**: Enter the password for the SYS user.
   - **Role**: `SYSDBA`

   Click **Next**.

6. If you get this warning message: The database you are connecting is with non-UTF8 charset, if you are going to use this database for multilingual support, you may have data loss. If you are not using for multilingual support you can continue, otherwise we strongly recommend using UTF-8 database.

   Click **Ignore** or **Stop**.

7. In the Select Components screen, do the following:

   - Select **Create a New Prefix**, and enter a prefix to use for the database schemas. Example: `DEV` or `PROD`. Prefixes are used to create logical groupings of multiple repositories in a database. For more information, see *Oracle Fusion Middleware Repository Creation Utility User's Guide*.

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

- Select the following:
  - AS Common Schemas:

    - **Metadata Services**
  - SOA Infrastructure:

    - **SOA and BPM Infrastructure**

    - **User Messaging Service**

    - **Business Activity Monitoring**

> **Note:** Business Activity Monitoring (BAM) is only required for BAM installations as described in Chapter 7, "Extending the Domain to Include BAM."

Click **Next**.

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

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

2.1.3 Configuring SOA Schemas for Transactional Recovery Privileges

You need the appropriate database privileges to allow the Oracle WebLogic Server transaction manager to query for transaction state information and issue the appropriate commands, such as commit and rollback, during recovery of in-flight transactions after a WebLogic Server container crash.

To configure the SOA schemas for transactional recovery privileges:

1. Log on to sqlplus as a user with sysdba privileges. For example:

   ```
   sqlplus "/ as sysdba"
   ```

2. Enter the following commands:

   ```
   SQL> Grant select on sys.dba_pending_transactions to soa_schema_prefix;

   Grant succeeded.

   SQL> Grant force any transaction to soa_schema_prefix;
   ```

```
Grant succeeded.

SQL>
```

Note: These privileges should be granted to the owner of the soainfra schema, as determined by the RCU operations.

2.1.4 Backing Up the Database

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

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

2.2 Network

This section covers the following topics:

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

2.2.1 Virtual Server Names

The SOA enterprise topology uses the following virtual server names:

- Section 2.2.1.1, "soa.mycompany.com"
- Section 2.2.1.2, "admin.mycompany.com"
- Section 2.2.1.3, "soainternal.mycompany.com"

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

2.2.1.1 soa.mycompany.com

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

2.2.1.2 admin.mycompany.com

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

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

This virtual server is defined on the load balancer.

2.2.1.3 soainternal.mycompany.com

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

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

This virtual server is defined on the load balancer.

2.2.2 Load Balancers

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

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

2.2.2.1 Configuring the Load Balancer

To configure the load balancer, complete these steps:

1. Create a pool of servers. You will assign this pool to virtual servers.
2. Add the addresses of the Oracle HTTP Server hosts to the pool. For example:
   - WEBHOST1:7777
   - WEBHOST2:7777
3. Configure a virtual server in the load balancer for `soa.mycompany.com:443`.
   - For this virtual server, use your system's frontend address as the virtual server address (for example, `soa.mycompany.com`). The frontend address is the externally facing host name used by your system and that will be exposed in the Internet.
   - Configure this virtual server with port 80 and port 443. Any request that goes to port 80 should be redirected to port 443.
   - Specify ANY as the protocol (non-HTTP protocols are required for B2B).
   - Enable address and port translation.
   - Enable reset of connections when services and/or nodes are down.
   - Assign the pool created in step 1 to the virtual server.
   - Create rules to filter out access to `/console` and `/em` on this virtual server.

4. Configure a virtual server in the load balancer for `admin.mycompany.com:80`.
   - For this virtual server, use your internal administration address as the virtual server address (for example, `admin.mycompany.com`). This address is typically not externalized.
   - Specify HTTP as the protocol.
   - Enable address and port translation.
   - Enable reset of connections when services and/or nodes are down.
   - Assign the pool created in step 1 to the virtual server.
5. Configure a virtual server in the load balancer for `soainternal.mycompany.com:80`.
   - For this virtual server, use your internal administration address as the virtual server address (for example, `soainternal.mycompany.com`). This address is typically not externalized.
   - Specify HTTP as the protocol.
   - Enable address and port translation.
   - Enable reset of connections when services and/or nodes are down.
   - Assign the pool created in step 1 to the virtual server.
   - Optionally, create rules to filter out access to `/console` and `/em` on this virtual server.
6. Configure 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 time response 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 2–1. As shown in this figure, each VIP and IP is attached to the WebLogic server that uses it. VIP1 is failed manually to restart the Administration Server in SOAHOST2. VIP2 and VIP3 fail over from SOAHOST1 to SOAHOST2 and from SOAHOST2 to SOAHOST1 respectively through Oracle WebLogic Server Migration feature. WLS_BAM1 also uses server migration to failover VIP4 from BAMHOST1 to BAMHOST2. See *Oracle Fusion Middleware High Availability Guide* for information on the WebLogic Server Migration feature. Physical IPs (non virtual) are fixed to each node. IP1 is the physical IP of SOAHOST1 and is used by the WLS_WSM1 WebServices Policy Manager server. IP2 is the physical IP of SOAHOST2 and is used by the WLS_WSM2 WebServices Policy Manager server. IP3 is the physical IP of BAMHOST2 and is used as the listen address by the WLS_BAM2 Server.

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

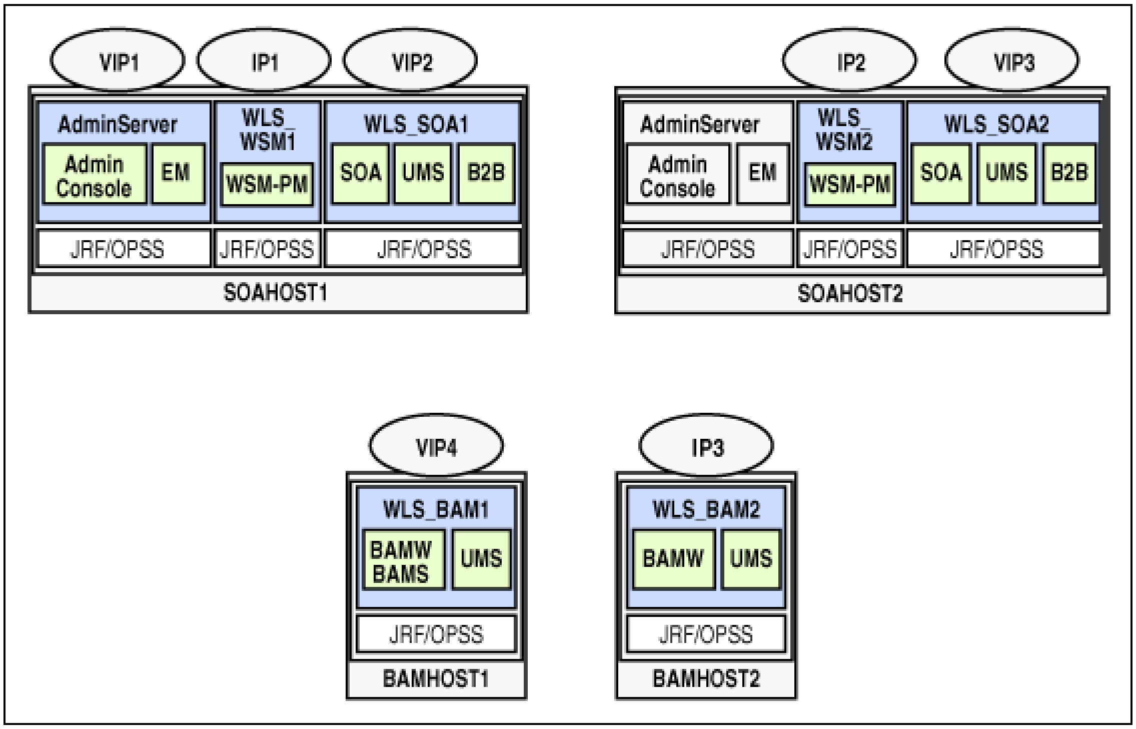


Table 2–2 provides descriptions of the various virtual hosts.

Table 2–2 Virtual Hosts

Virtual IP	VIP Maps to...	Description
VIP1	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	BAMHOST1VHN1	BAMHOST1VHN1 is the virtual host name that maps to the listen address for WLS_BAM1 and fails over with server migration of this managed server. It is enabled on the node where WLS_BAM1 process is running (BAMHOST1 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 SOA 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	HTTPS / 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	HTTPS / 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 2.2.2.1, "Configuring the Load Balancer."
OHS registration with Administration Server	FW1	7001	HTTP/t3	Inbound	Set the timeout to a short period (5-10 seconds).
OHS management by Administration Server	FW1	OPMN port (6701) and OHS Admin Port (7779)	TCP and HTTP, respectively	Outbound	Set the timeout to a short period (5-10 seconds).

Table 2–3 (Cont.) Ports Used

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
WSM-PM access	FW1	7010 Range: 7010 - 7999	HTTP / WLS_ WSM-PM*n*	Inbound	Set the timeout to 60 seconds.
SOA Server access	FW1	8001 Range: 8000 - 8080	HTTP / WLS_SOA*n*	Inbound	Timeout varies based on the type of process model used for SOA.
BAM access	FW1	9001 Range: 9000 - 9080	HTTP / WLS_ BAM*n*	Inbound	Connections to BAM WebApps are kept open until the report/browser is closed, so set the timeout as high as the longest expected user session.
Communication between SOA Cluster members	n/a	8001	TCP/IP Unicast	n/a	By default, this communication uses the same port as the server's listen address.
Communication between WSM Cluster members	n/a	7010	TCP/IP Unicast	n/a	By default, this communication uses the same port as the server's listen address.
Session replication within a WebLogic Server cluster	n/a	n/a	n/a	n/a	By default, this communication uses the same port as the server's listen address.
Administration Console access	FW1	7001	HTTP / Administration Server and Enterprise Manager t3	Both	You should tune this timeout based on the type of access to the admin console (whether it is planned to use the Oracle WebLogic Server Administration Console from application tier clients or clients external to the application tier).
Node Manager	n/a	5556	TCP/IP	n/a	n/a For actual values, see "Firewalls and Ports" in *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	

Table 2–3 (Cont.) Ports Used

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Database access	FW2	1521	SQL*Net	Both	Timeout depends on all database content and on the type of process model used for SOA.
Coherence for deployment	n/a	8088 Range: 8000 - 8090		n/a	n/a
Oracle Internet Directory access	FW2	389	LDAP	Inbound	You should tune the directory server's parameters based on load balancer, and not the other way around.
Oracle Internet Directory access	FW2	636	LDAP SSL	Inbound	You should tune the directory server's parameters based on load balancer, and not the other way around.
JOC for OWSM	n/a	9991	TCP/IP	n/a	n/a

Note: The TCP/IP port for B2B is a user-configured port and is not predefined. Similarly, 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 an Enterprise Deployment topology. Other directory layouts are possible and supported, but the model adopted in this guide is chosen for maximum availability, providing both the best isolation of components and symmetry in the configuration and facilitating backup and disaster recovery. The rest of the document uses this directory structure and directory terminology.

This section covers these topics:

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

2.3.1 Terminology for Directories and Directory Environment Variables

- **ORACLE_BASE:** This environment variable and related directory path refers to the base directory under which Oracle products are installed.
- **MW_HOME:** This environment variable and related directory path refers to the location where Fusion Middleware (FMW) resides.
- **WL_HOME:** This environment variable and related directory path contains installed files necessary to host a WebLogic Server.

- **ORACLE_HOME:** This environment variable and related directory path refers to the location where Oracle FMW SOA Suite is installed.
- **ORACLE_COMMON_HOME**: This environment variable and related directory path refers to the Oracle home that contains the binary and library files required for the Oracle Enterprise Manager Fusion Middleware Control and Java Required Files (JRF).
- **DOMAIN Directory:** This directory path refers to the location where the Oracle WebLogic Domain information (configuration artifacts) is stored. Different WLS Servers can use different domain directories even when in the same node as described below.
- **ORACLE_INSTANCE:** An Oracle instance contains one or more system components, such as Oracle Web Cache, Oracle HTTP Server, or Oracle Internet Directory. An Oracle instance directory contains updateable files, such as configuration files, log files, and temporary files.

2.3.2 Recommended Locations for the Different Directories

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

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

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

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)

> **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.

- Mounted from: Nodes alternatively mount VOL1 or VOL2 so that at least half of the nodes use one installation, and half use the other.

  In a SOA Enterprise Deployment topology, SOAHOST1 mounts VOL1 and SOAHOST2 mounts VOL2. When only one volume is available, nodes mount the two suggested directories in shared storage alternately. For example, SOAHOST1 would use *ORACLE_BASE*/product/fmw1 as a shared storage location, and SOAHOST2 would use *ORACLE_BASE*/product/fmw2 as a shared storage location)

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)

> **Note:** Web Tier installation is typically performed on local storage to the WEBHOST nodes. When using shared storage, consider the appropriate security restrictions for access to the storage device across tiers.
>
> This enterprise deployment guide assumes that the Oracle Web Tier will be installed onto local disk. You may install the Oracle Web Tier binaries (and the ORACLE_INSTANCE) onto shared disk. If so, the shared disk MUST be separate from the shared disk used for the application tier.

- Mounted from: For Shared Storage installations, nodes alternatively mount VOL1 or VOL2 so that at least half of the nodes use one installation, and half use the other.

  In a SOA Enterprise Deployment topology, WEBHOST1 mounts VOL1 and WEBHOST2 mounts VOL2. When only one volume is available, nodes mount the two suggested directories in shared storage alternately. For example, WEBHOST1 would use *ORACLE_BASE*/product/fmw1 as a shared storage location, and WEBHOST2 would use *ORACLE_BASE*/product/fmw2 as a shared storage location).

WL_HOME:

MW_HOME/wlserver_10.3

ORACLE_HOME:

MW_HOME/soa

ORACLE_COMMON_HOME:

MW_HOME/oracle_common

ORACLE_INSTANCE:

ORACLE_BASE/admin/*instance_name*

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

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

Domain Directory for Administration Server Domain Directory:

ORACLE_BASE/admin/*domain_name*/aserver/*domain_name* (The last “domain_name” is added by config wizard)

- Mount point on machine: *ORACLE_BASE*/admin/*domain_name*/aserver
- Shared storage location: *ORACLE_BASE*/admin/*domain_name*/aserver
- 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 Domain Directory:

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

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

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

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

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

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

- Mount point: *ORACLE_BASE*/admin/*domain_name*/*soa_cluster_name*/
- Shared storage location: *ORACLE_BASE*/admin/*domain_name*/*soa_cluster_name*/
- Mounted from: All nodes running SOA or BAM must mount this shared storage location so that transaction logs and JMS stores are available when server migration to another node take place.

Location for Application Directory for the Administration Server

ORACLE_BASE/admin/*domain_name*/aserver/applications

- Mount point: *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 must 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

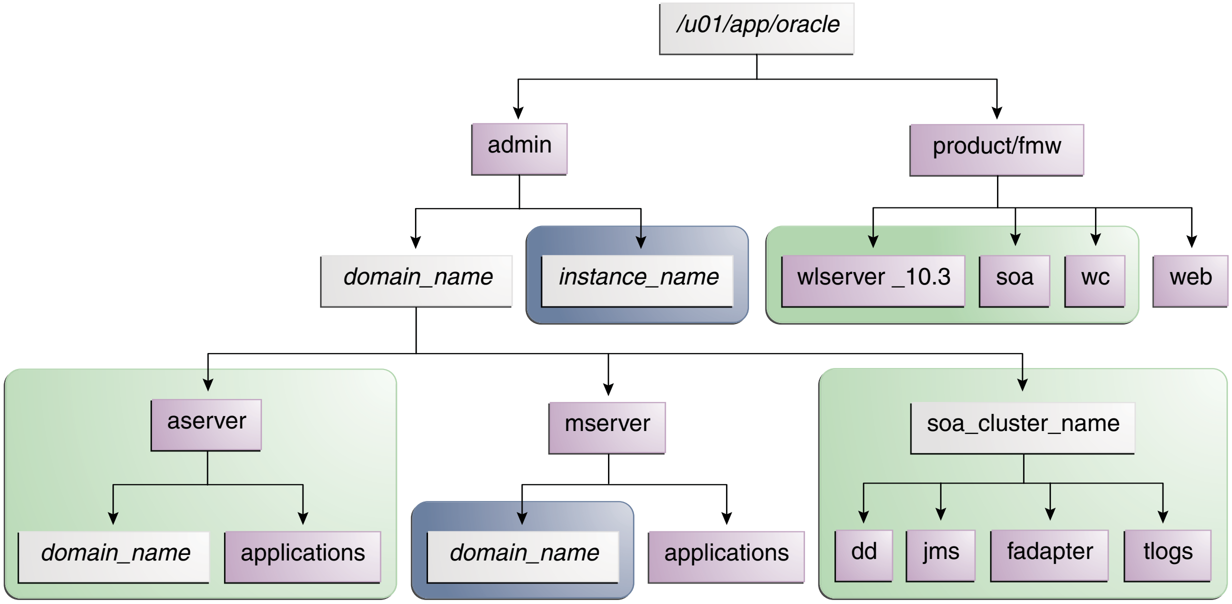
Location for Application Directory for Managed Server

ORACLE_BASE/admin/*domain_name*/mserver/applications

> **Note:** This directory is local in the context of a SOA enterprise deployment.

Figure 2–2 illustrates this directory structure

Figure 2–2 Directory Structure



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

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

Table 2–4 Directory Structure Elements

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

Figure 2–3 shows an example configuration for shared storage with multiple volumes for SOA. This can be extrapolated with the same structure for BAM deployments.

Figure 2–3 Example Configuration for Shared Storage

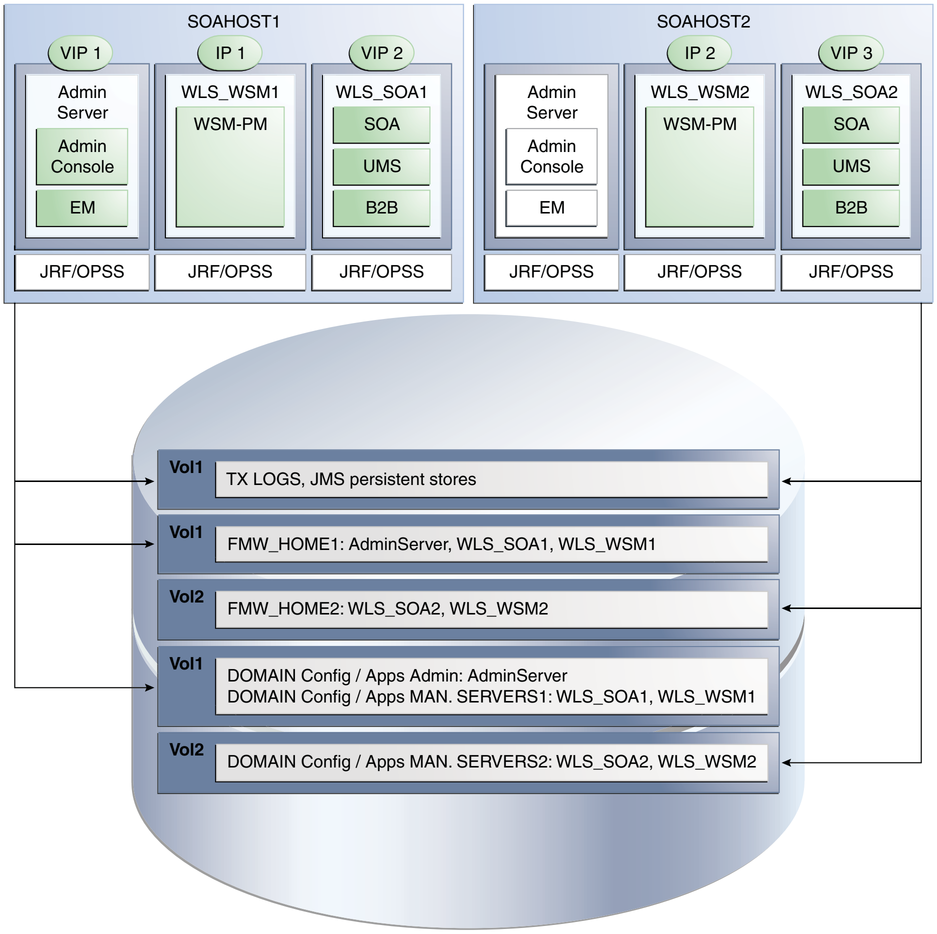

SOAHOST1
VIP 1
IP 1
VIP 2
Admin Server
Admin Console
EM
WLS_WSM1
WSM-PM
WLS_SOA1
SOA
UMS
B2B
JRF/OPSS
JRF/OPSS
JRF/OPSS
SOAHOST2
IP 2
VIP 3
Admin Server
Admin Console
EM
WLS_WSM2
WSM-PM
WLS_SOA2
SOA
UMS
B2B
JRF/OPSS
JRF/OPSS
JRF/OPSS
Vol1
TX LOGS, JMS persistent stores
Vol1
FMW_HOME1: AdminServer, WLS_SOA1, WLS_WSM1
Vol2
FMW_HOME2: WLS_SOA2, WLS_WSM2
Vol1
DOMAIN Config / Apps Admin: AdminServer
DOMAIN Config / Apps MAN. SERVERS1: WLS_SOA1, WLS_WSM1
Vol2
DOMAIN Config / Apps MAN. SERVERS2: WLS_SOA2, WLS_WSM2

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.

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

Server	Type of Data	Volume in Shared Storage	Directory	Files
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>	For configurations where managed server domain directory is located in shared storage, the files would be individual in each volume, but both servers see the same directory structure. In this guide, local storage is used for managed server domain directories.
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/vol1/u01/app/oracle/product/fmw
/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 depending on the specific storage device.

```
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.

Note: The user ID used to create a shared storage file system owns and has read, write, and execute privileges for those files. Other users in the operating system group can read and execute the files, but they do not have write privileges. For more information about installation and configuration privileges, see section 3.3, "Understanding Installation and Configuration Privileges and Users" in the *Oracle Fusion Middleware Installation Planning Guide*.

2.4 LDAP as Credential and Policy Store

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

An Oracle Fusion Middleware SOA Suite Enterprise Deployment Topology uses different domain homes for the Administration Server and the managed server as described in the Section 2.3, "Shared Storage and Recommended Directory Structure." Derived from this, and for integrity and consistency purposes, Oracle requires the use of an LDAP as policy and credential store in context of Oracle Fusion Middleware SOA Suite Enterprise Deployment Topology. To configure the Oracle Fusion Middleware SOA Suite Enterprise Deployment Topology with an LDAP as Credential and Policy store, follow the steps in Section 11.1, "Credential and Policy Store Configuration."

3 Installing Oracle HTTP Server

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

This chapter contains the following sections:

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

3.1 Installing Oracle HTTP Server on WEBHOST1 and WEBHOST2

As described in Section 2.3, "Shared Storage and Recommended Directory Structure," Oracle recommends installing Oracle Web Tier onto local disk. You may, however, install the Web Tier onto shared disk, if this is the case:

- Ensure the software is installed in at least two storage locations for redundancy.
- The shared storage is different to that which is used for the application tier.

To install Oracle HTTP Server on WEBHOST1 and WEBHOST2:

1. Check that your machines meet the following requirements:

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

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

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

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

3. Start the Oracle Universal Installer from the Oracle Fusion Middleware 11*g* WebTier and Utilities DVD by issuing this command:

    ```
    runInstaller
    ```

4. If the Specify Inventory Directory screen appears, enter the location for the inventory and the user group and then click **OK**; otherwise, ignore steps 4 and 5 and continue with step 6.

5. Execute the root privileged actions as indicated in the dialog and then click **OK**.

6. In the **Welcome** screen, click **Next**.
7. In the **Select Installation Type** screen, select **Install and Configure**, and click **Next**.
8. In the **Prerequisite Checks** screen, ensure that all the prerequisites are met, then click **Next**.
9. In the **Specify Installation Location** screen, do the following:
   - On WEBHOST1, set the **Location** to:

     ```
     MW_HOME/web
     ```

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

     ```
     MW_HOME/web
     ```

   Click **Next**.
10. In the **Configure Components** screen, do the following:
    - Select **Oracle Http Server**.
    - Do *not* select **Oracle Web Cache**.
    - Do *not* select **Associate Selected Components with WebLogic Domain** because you have not yet installed WebLogic Server.

    Click **Next**.
11. In the **Specify Component Details** screen, do the following:
    - Enter the following values for WEBHOST1:
      - **Instance Home Location**: `ORACLE_BASE/admin/<instance_name>`
      - **Instance Name**: `ohs_instance1`
      - **OHS Component Name**: `ohs1`
    - Enter the following values for WEBHOST2:
      - **Instance Home Location**: `ORACLE_BASE/admin/<instance_name>`
      - **Instance Name**: `ohs_instance2`
      - **OHS Component Name**: `ohs2`

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

    Click **Next**.

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

13. In the **Specify Security Updates** screen, enter your e-mail address to receive e-mail notifications of security issues (if required). Enter your Oracle Support Password to receive security updates through My Oracle Support.
14. In the **Installation Summary** screen, ensure that the selections are correct, and click **Install**.
15. In the **Configuration** screen, multiple configuration assistants are launched in succession, which can be a lengthy process. When it completes, the **Configuration Completed** screen appears.
16. In the **Installation Completed** screen, click **Finish** to exit.

3.2 Validating Oracle HTTP Server Through the Load Balancer

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

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

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

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

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

```
WEBHOST> cd ORACLE_BASE/admin/<instance_name>/bin
WEBHOST> opmnctl stopall
WEBHOST> opmnctl startall
```

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

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

3.3 Backing Up Oracle HTTP Server

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

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

1. Shut down the instance using `opmnctl`.

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

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

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

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

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

4. Start the instance using `opmnctl`:

   ```
   ORACLE_BASE/admin/instance_namebin/opmnctl startall
   ```

4 Creating a Domain

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

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

This chapter contains the following sections:

- Section 4.16, "Starting and Validating the WLS_WSM2 Managed Server"
- Section 4.17, "Configuring the Java Object Cache for Oracle WSM"
- Section 4.18, "Configuring Oracle HTTP Server for the Administration Server and the WLS_WSM-PMn Managed Servers"
- Section 4.19, "Registering Oracle HTTP Server With WebLogic Server"
- Section 4.20, "Setting the Frontend URL for the Administration Console and Setting Redirection Preferences"
- Section 4.21, "Validating Access Through Oracle HTTP Server"
- Section 4.22, "Manually Failing Over the Administration Server to SOAHOST2"
- Section 4.23, "Validating Access to SOAHOST2 Through Oracle HTTP Server"
- Section 4.24, "Failing the Administration Server Back to SOAHOST1"
- Section 4.25, "Backing Up the Installation"

4.1 Installing Oracle Fusion Middleware Home

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

You must install the following components of Oracle Fusion Middleware:

- Oracle WebLogic Server (see Section 4.1.1, "Installing Oracle WebLogic Server")
- Oracle SOA Suite (see Section 4.1.2, "Installing Oracle Fusion Middleware for SOA")

4.1.1 Installing Oracle WebLogic Server

Install Oracle WebLogic Server on SOAHOST1 and SOAHOST2.

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*.

Oracle WebLogic Server on SOAHOST1 and SOAHOST2.

To install Oracle WebLogic Server:

1. Start the Oracle WebLogic Server installer.

   > **Note:** For information about the correct WebLogic Server installation program file name, see the section "Starting the Installation Program in Graphical Mode," in the *Oracle Fusion Middleware Installation Guide for Oracle WebLogic Server*.

   For UNIX (Linux used in this example):

   ```
   SOAHOST1> wls_linux32.bin
   ```

   For Windows operating systems:

   ```
   SOAHOST1> wls_win32.exe
   ```

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

3. In the Choose Middleware Home Directory screen, do the following:

   - Select **Create a New Middleware Home**.
   - For Middleware Home Directory, enter **MW_HOME**.

   > **Note:** 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 directory **WL_HOME**, and click **Next**.
9. In the Installation Summary screen, click **Next**.
10. In the Installation Complete screen, unselect **Run QuickStart**, and click **Done**.

4.1.2 Installing Oracle Fusion Middleware for SOA

Perform these steps to install Oracle Fusion Middleware for SOA on SOAHOST1 and SOAHOST2.

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

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

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

   ```
   SOAHOST1> runInstaller
   ```

   When the installer prompts you for a JRE/JDK location, enter the Oracle SDK location created in the Oracle WebLogic Server installation, for example, **MW_HOME/jrockit_160_<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 **Next**.

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

4. In the Welcome screen, click **Next**.
5. In the Prerequisite Check screen, verify that the checks complete successfully, and click **Next**.

6. Specify the installation location. Select the previously installed Middleware Home from the drop-down list. For the Oracle Home directory, enter the directory name (**soa**).

   Click **Next**.

7. In the Application Server screen, select **WebLogic**.

   Click **Next**.

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

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

> **Note:** This Enterprise Deployment Guide assumes that the binary installation for both Oracle Fusion Middleware Components and Oracle HTTP Server are patched to the latest patch set available. The Enterprise Deployment Guide 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 patch set before the Configuration Wizard and domain operations are started. Follow the documentation included with each patch set to properly patch an existing installation. Specifically for Oracle BPM, Oracle requires the WL_HOME and ORACLE_HOME to be patched to the latest Oracle Fusion Middleware patch set level before the WebLogic Configuration Wizard steps are performed to extend a domain. For information about the latest versions and patching procedures for Oracle Fusion Middleware, see the *Oracle Fusion Middleware Patching Guide*.

4.2 Backing Up the Installation

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

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

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

4.3 Enabling VIP1 in SOAHOST1

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

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

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

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

For example:

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

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

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

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

```
/bin/ping 100.200.140.206
```

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

4.4 Running the Configuration Wizard on SOAHOST1 to Create a Domain

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

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 directory to the location of the Configuration Wizard. This is within the SOA home directory.

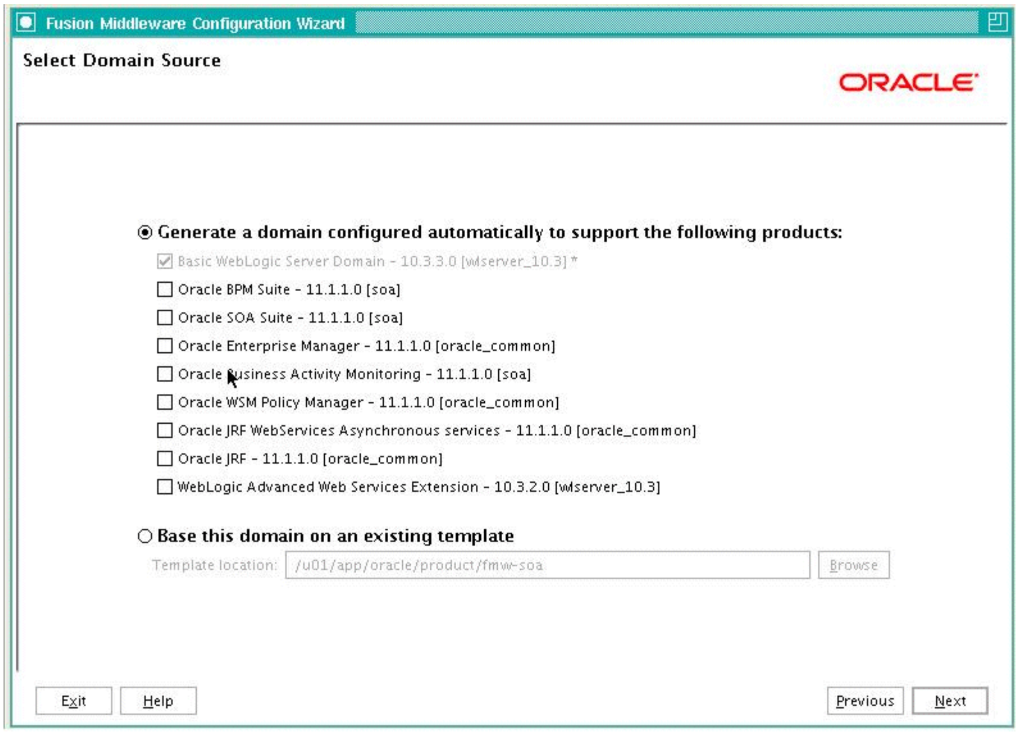
   ```
   SOAHOST1> cd ORACLE_COMMON_HOME/common/bin
   ```

3. Start the Oracle Fusion Middleware Configuration Wizard:

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

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

Figure 4–1 Select Domain Source Screen



Select Domain Source

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

☑ Basic WebLogic Server Domain - 10.3.3.0 [wlserver_10.3] *

□ Oracle BPM Suite - 11.1.1.0 [soa]

□ Oracle SOA Suite - 11.1.1.0 [soa]

□ Oracle Enterprise Manager - 11.1.1.0 [oracle_common]

□ Oracle Business Activity Monitoring - 11.1.1.0 [soa]

□ Oracle WSM Policy Manager - 11.1.1.0 [oracle_common]

□ Oracle JRF WebServices Asynchronous services - 11.1.1.0 [oracle_common]

□ Oracle JRF - 11.1.1.0 [oracle_common]

□ WebLogic Advanced Web Services Extension - 10.3.2.0 [wlserver_10.3]

○ Base this domain on an existing template

Template location: /u01/app/oracle/product/fmw-soa Browse

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

****

In the Select Domain Source screen, do the following:

- Select **Generate a domain configured automatically to support the following products**.
- Select the following products:
  - **Basic WebLogic Server Domain - 10.3.4.0 [wlserver_10.3]** (this should be selected automatically)
  - **Oracle Enterprise Manager - 11.1.1.0 [oracle_common]**
  - **Oracle WSM Policy Manager 11.1.1.0 [oracle_common]**
  - **Oracle JRF - 11.1.1.0 [oracle_common]** (this should be selected automatically)

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

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

Click **Next**.

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

Make sure that the domain directory matches the directory and shared storage mount point recommended in Chapter 2, "Database and Environment Preconfiguration": enter `ORACLE_BASE/admin/<domain_name>/aserver/` for the domain directory and `ORACLE_BASE/admin/<domain_name>/aserver/applications` for the application directory. This directory should be in shared storage.

7. Click **Next**.

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

   Click **Next**.

9. In the Configure Server Start Mode and JDK screen, do the following:

   - For WebLogic Domain Startup Mode, select **Production Mode**.
   - For JDK Selection, select **JROCKIT SDK1.6.0_<version>**.

   Click **Next**.

10. In the Configure JDBC Components Schema screen, do the following:

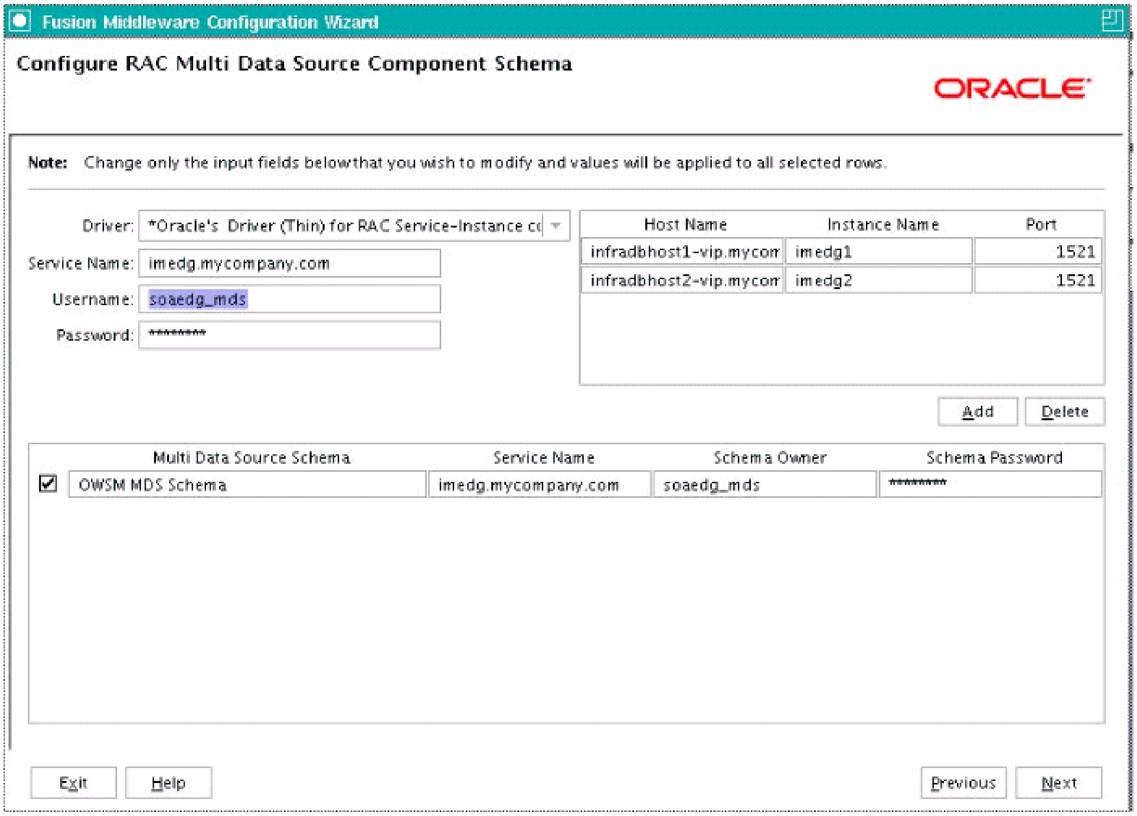
    a. Select the OWSM MDS schema.

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

    c. Click **Next**.

11. The Configure RAC Multi Data Sources Component Schema screen is displayed (Figure 4–2).

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



Fusion Middleware Configuration Wizard

Configure RAC Multi Data Source Component Schema

Note: Change only the input fields belowthat 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: imedg.mycompany.com

Username: soaedg_mds

Password: *********

Host Name	Instance Name	Port
infradbhost1-vip.mycom	imedg1	1521
infradbhost2-vip.mycom	imedg2	1521

	Multi Data Source Schema	Service Name	Schema Owner	Schema Password
☑	OWSM MDS Schema	imedg.mycompany.com	soaedg_mds	*********

In this screen, do the following:

a. Enter values for the following fields, specifying the connect information for the Oracle RAC database that was seeded with RCU.

   – **Driver:** Select **Oracle driver (Thin) for RAC Service-Instance connections, Versions:10, 11**.

   – **Service Name:** Enter the service name of the database, for example, `soaedg.mycompany.com`.

   – **Username:** Enter the complete user name (including the prefix) for the schemas.

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

b. Enter the host name, instance name, and port.

c. Click **Add**.

d. Repeat this for each Oracle RAC instance.

e. Click **Next**.

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

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

13. In the Select Advanced Configuration screen, select the following:

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

    Click **Next**.

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

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

    Click **Next**.

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

Table 4–1 Managed Servers

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

    Click **Next**.

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

Table 4–2 Clusters

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

Click **Next**.

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

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

    Click **Next**.

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

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

      Note: "Name" can be any unique string. "Node Manager Listen Address" must be a resolvable host name.

Table 4–3 Machines

Name	Node Manager Listen Address
SOAHOST1	SOAHOST1
SOAHOST2	SOAHOST2
ADMINHOST	localhost

Leave all other fields to their default values.

Click **Next**.

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

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

    Click **Next**.

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

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

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

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

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

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

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

22. In the Configuration Summary screen, click **Create**.
23. In the Create Domain screen, click **Done**.

4.5 Creating boot.properties for the Administration Server on SOAHOST1

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

For the Administration Server:

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 last directory created in the previous step, and enter the following lines in the file:

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

> **Note:** When you start the Administration Server, the username and password entries in the file get encrypted. You start the Administration Server in Section 4.7, "Starting the Administration Server on SOAHOST1."
>
> For security reasons, you want to minimize the time the entries in the file are left unencrypted: after you edit the file, you should start the server as soon as possible so that the entries get encrypted.

4.6 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 10.7.5, "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 you set `-DDomainRegistrationEnabled=true` whenever a Node Manager is started, that must manage the AdminServer. If there is no AdminServer on this machine and this machine is not an AdminServer failover node, you can start the Node Manager as:
>
> ```
> SOAHOST1> ./startNodeManager.sh
> ```

4.7 Starting the Administration Server on SOAHOST1

The Administration Server is started and stopped using Node Manager. However, the first start of the Administration Server with Node Manager, requires changing the defaulted username and password that are set for Node Manager by the Configuration Wizard. 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 1-4 are required for the first start operation, 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. In a browser, go to the following URL;

      `http://ADMINVHN:7001/console`

   b. Log in as the administrator.

   c. Click **Lock and Edit**.

   d. Click **domain_name**, **Security**, **General**, and then expand the **Advanced** options at the bottom.

   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 by using **CTRL-C** in the shell where it was started, or by process identification and kill in the OS.

4. Start WLST and connect to Node Manager with **nmconnect** and the credentials set in the previous steps and start the Administration Server using **nmstart**.

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

Once you are in the WLST shell:

```
wls:/offline>nmConnect('Admin_User','Admin_Password',
'SOAHOST1','5556','domain_name','/u01/app/oracle/admin/domain_
name/aserver/domain_name')

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

Note: This username and password are used only to authenticate connections between Node Manager and clients. They are independent of 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.

4.8 Validating the Administration Server

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

1. In a browser, go to `http://ADMINVHN:7001/console`.
2. Log in as the administrator.
3. Verify that the WLS_WSM1 and WLS_WSM2 managed servers are listed.
4. Verify that the WSM-PM_Cluster cluster is listed.
5. Check that you can access Oracle Enterprise Manager at `http://ADMINVHN:7001/em`.
6. Log in to EM Console with the username and password you specified in Section 4.5, "Creating boot.properties for the Administration Server on SOAHOST1."

4.9 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 SOAHOST1as recommended in Chapter 2, "Database and Environment Preconfiguration."

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

```
SOAHOST1> cd ORACLE_COMMON_HOME/common/bin

SOAHOST1> ./pack.sh -managed=true -domain=ORACLE_BASE/admin/domain_
name/aserver/domain_name -template=soadomaintemplate.jar -template_name=soa_
domain_template
```

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

```
SOAHOST1> cd ORACLE_COMMON_HOME/common/bin

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

```
name/mserver/applications
```

4.10 Applying the Java Required Files (JRF) Template to the WSM-PM_ Cluster

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

To target these resources:

1. Log in to Oracle Enterprise Manager Fusion Middleware Control with the username and password you specified in Section 4.5, "Creating boot.properties for the Administration Server on SOAHOST1."
2. On the navigation tree on the left, expand **Farm_<domain_name>**, **WebLogic Domain**, and then **<domain_name>**, and select **WSM_PM_Cluster**.
3. Click **Apply JRF Template** on the right.
4. Wait for the confirmation message to appear on the screen.

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

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

This step is required if you have not set up the appropriate certificates to authenticate the different nodes with the Administration Server (see Chapter 8, "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 Enterprise Deployment topology configuration is complete as described in Chapter 8, "Setting Up Node Manager."

Perform these steps to disable host name verification:

1. Log in to Oracle WebLogic Server Administration Console.
2. Click **Lock and Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page appears.
5. Select **AdminServer(admin)** in the Names column of the table. The Settings page for AdminServer(admin) appear.
6. Click the **SSL** tab.
7. Click **Advanced**.
8. Set Hostname Verification to **None**.
9. Click **Save**.
10. Repeat steps 4 to 8 for the WLS_WSM1 server.
11. Save and activate the changes.
12. The change requires restart of the Administration Server to be effective. To do this, complete these steps:

    a. In the Summary of Servers screen, select the **Control** tab.

b. Select **AdminServer(admin)** in the table and then click **Shutdown**.

c. Start the Administration Server again using the procedure in Section 4.7, "Starting the Administration Server on SOAHOST1."

4.12 Starting and Validating the WLS_WSM1 Managed Server

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

1. Start the WLS_WSM1 managed server using the Oracle WebLogic Server Administration Console as follows:

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

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

   c. Click the **Control** tab.

   d. Select **WLS_WSM1** and then click **Start**.

2. Verify that the server status is reported as "Running" in the Admin Console. If the server is shown as "Starting" or "Resuming," wait for the server status to change to "Started." If another status is reported (such as "Admin" or "Failed"), check the server output log files for errors. See Section 10.7, "Troubleshooting" for possible causes.

3. Access `http://SOAHOST1:7010/wsm-pm`.

4. Click **Validate Policy Manager**. A list of policies and assertion templates available in the data store appear.

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

4.13 Propagating the Domain Configuration to SOAHOST2 Using the unpack Utility

Perform these steps to propagate the domain configuration:

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

```
SOAHOST1> cd ORACLE_COMMON_HOME/common/bin
SOAHOST1> scp soadomaintemplate.jar oracle@SOAHOST2:/ORACLE_COMMON_
HOME/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 the `WL_HOME/common/bin` directory.

```
SOAHOST2> cd ORACLE_COMMON_HOME/common/bin

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

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

4.14 Disabling Host Name Verification for the WLS_WSM2 Managed Server

This step is required if you have not set up the appropriate certificates to authenticate the different nodes with the Administration Server (see Chapter 8, "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 Enterprise Deployment topology configuration is complete as described in Chapter 8, "Setting Up Node Manager."

Perform these steps to disable host name verification:

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

4.15 Starting Node Manager on SOAHOST2

Perform these steps to start Node Manager on SOAHOST2:

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

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

   > **Note:** You must use the StartScriptEnabled property to avoid class loading failures and other problems. See also Section 10.7.5, "Incomplete Policy Migration After Failed Restart of SOA Server."

2. Start Node Manager:

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

4.16 Starting and Validating the WLS_WSM2 Managed Server

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

1. Start the WLS_WSM2 managed server using the Administration Console.
2. Verify that the server status is reported as "Running" in the Admin Console. If the server is shown as "Starting" or "Resuming," wait for the server status to change to "Started." If another status is reported (such as "Admin" or "Failed"), check the server output log files for errors. See Section 10.7, "Troubleshooting" for possible causes.
3. Access `http://SOAHOST2:7010/wsm-pm`.
4. Click validate policy manager.

4.17 Configuring the Java Object Cache for Oracle WSM

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

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

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

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

Usage

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

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

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

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

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

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

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

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

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

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

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

- Configure JOC for all specified managed servers.

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

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

- Exclude JOC configuration for some managed servers.

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

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

- Disable the distribution mode for all managed servers.

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

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

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

4.18 Configuring Oracle HTTP Server for the Administration Server and the WLS_WSM-PM*n* Managed Servers

To enable Oracle HTTP Server to route to the Administration Server and the WSM-PM_Cluster, which contain the WLS_WSM-PM*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:

```
# The admin URLs should only be accessible via the admin virtual host

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

# Admin Server and EM
<Location /console>
    SetHandler weblogic-handler
    WebLogicHost ADMINVHN
    WeblogicPort 7001
</Location>

<Location /consolehelp>
    SetHandler weblogic-handler
    WebLogicHost ADMINVHN
    WeblogicPort 7001
</Location>

<Location /em>
    SetHandler weblogic-handler
    WebLogicHost ADMINVHN
    WeblogicPort 7001
</Location>

<VirtualHost>
# Virtual host entry for external https URL configured at the Load Balancer

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

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

# Virtual host entry for internal http URL

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

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

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
   ```

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

Some example scenarios:

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

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

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

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. To do this, follow these steps:

1. Log on to the Administration Console.
2. Click on the domain name in the navigation tree on the left.
3. Click on the **Web Applications** tab.
4. Click **Lock and Edit**.
5. Select the **WebLogic Plugin Enabled** check box
6. Save and activate the changes.

4.19 Registering Oracle HTTP Server With WebLogic Server

Once an Oracle WebLogic domain is created, the Oracle Web Tier can be linked to the domain. The advantages of doing this is that the Oracle Web Tier can be managed and monitored using the Oracle Fusion Middleware Console.

To associate the Oracle Web Tier with the WebLogic domain use the following commands:

```
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.

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

When you access the Oracle WebLogic Server Administration Console using a load balancer, changing the Administration Server's frontend URL is required so that the user's browser is redirected to the appropriate load balancer address. To change the Administration Server's frontend URL, complete these steps:

1. Log in to Oracle WebLogic Server Administration Console.
2. Click **Lock and Edit**.
3. Expand the Environment node in the Domain Structure window.
4. Click **Servers** to open the Summary of Servers page.
5. Select **Admin Server** in the Names column of the table. The Settings page for AdminServer(admin) appear.
6. Click the Protocols tab.
7. Click the HTTP tab.
8. Set the **Frontend Host** to **admin.mycompany.com** and the **Frontend HTTP Port** to **80** (modify accordingly if HTTPS is used for the admin URL).
9. Save and activate the changes.

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

Note: If you have any issues activating any configuration changes after modifying the Frontend Host and Port settings, then refer to Section 10.7.14, "Redirecting of Users to Login Screen After Activating Changes in Administration Console."

4.21 Validating Access Through Oracle HTTP Server

Verify that the server status is reported as "Running" in the Admin Console. If the server is shown as "Starting" or "Resuming," wait for the server status to change to "Started." If another status is reported (such as "Admin" or "Failed"), check the server output log files for errors. See Section 10.7, "Troubleshooting" for possible causes.

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

- `http://WEBHOST_node1:7777/wsm-pm`
- `http://WEBHOST_node2:7777/wsm-pm`

- `http://WEBHOST_node1:7777/console`
- `http://WEBHOST_node2:7777/console`
- `http://WEBHOST_node1:7777/em`
- `http://WEBHOST_node2:7777/em`
- `https://soa.mycompany.com/wsm-pm`
- `http://admin.mycompany.com/console`
- `http://admin.mycompany.com/em`

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

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

4.22 Manually Failing Over the Administration Server to SOAHOST2

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

Assumptions:

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

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

1. Stop the Administration Server.
2. Migrate IP to the second node.
   a. Run the following command as root on SOAHOST1 (where X:Y is the current interface used by 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 10.0.0.1 netmask 255.255.255.0
```

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

3. Update routing tables through `arping`, for example:

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

4. Start the Administration Server on SOAHOST2 using the procedure in Section 4.7, "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`.

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

4.23 Validating Access to SOAHOST2 Through Oracle HTTP Server

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

4.24 Failing the Administration Server Back to SOAHOST1

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

1. Run the following command on SOAHOST2.

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

2. Run the following command on SOAHOST1:

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

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

---

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

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

4. Start the Administration Server again on SOAHOST1 using the procedure in Section 4.7, "Starting the Administration Server on SOAHOST1.".

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

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

4.25 Backing Up the Installation

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

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

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

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

5 Extending the Domain for SOA Components

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

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

This chapter contains the following sections:

5.1 Enabling VIP2 on SOAHOST1 and VIP3 on SOAHOST2

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

To enable the VIP, follow the steps described in Section 4.3, "Enabling VIP1 in SOAHOST1." These VIPs and VHNs are required to enable server migration for the SOA Servers. Server migration must be configured for the SOA System for high availability purposes. Refer to Chapter 9, "Server Migration" for more details on configuring server migration for the SOA servers.

5.2 Extending the Domain for SOA Components

In this step, you extend the domain created in Chapter 4, "Creating a Domain" to contain SOA components.

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

Note: Oracle SOA uses Quartz to maintain its jobs and schedules in the database. The system clocks for the SOA WebLogic cluster must be synchronized to enable proper functioning of jobs, adapters, and Oracle B2B.

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

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

2. Start the Configuration Wizard.

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

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

4. In the WebLogic Domain Directory screen, select the WebLogic domain directory (`ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>`), and click **Next**.
5. In the Select Extension Source screen, do the following:
   - Select **Extend my domain automatically to support the following added products**.
   - Select the following products:
     – **Oracle SOA Suite 11.1.1.0**

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

     – Basic WebLogic Server Domain

     – Oracle Enterprise Manager

     – Oracle WSM Policy Manager

     – Oracle JRF

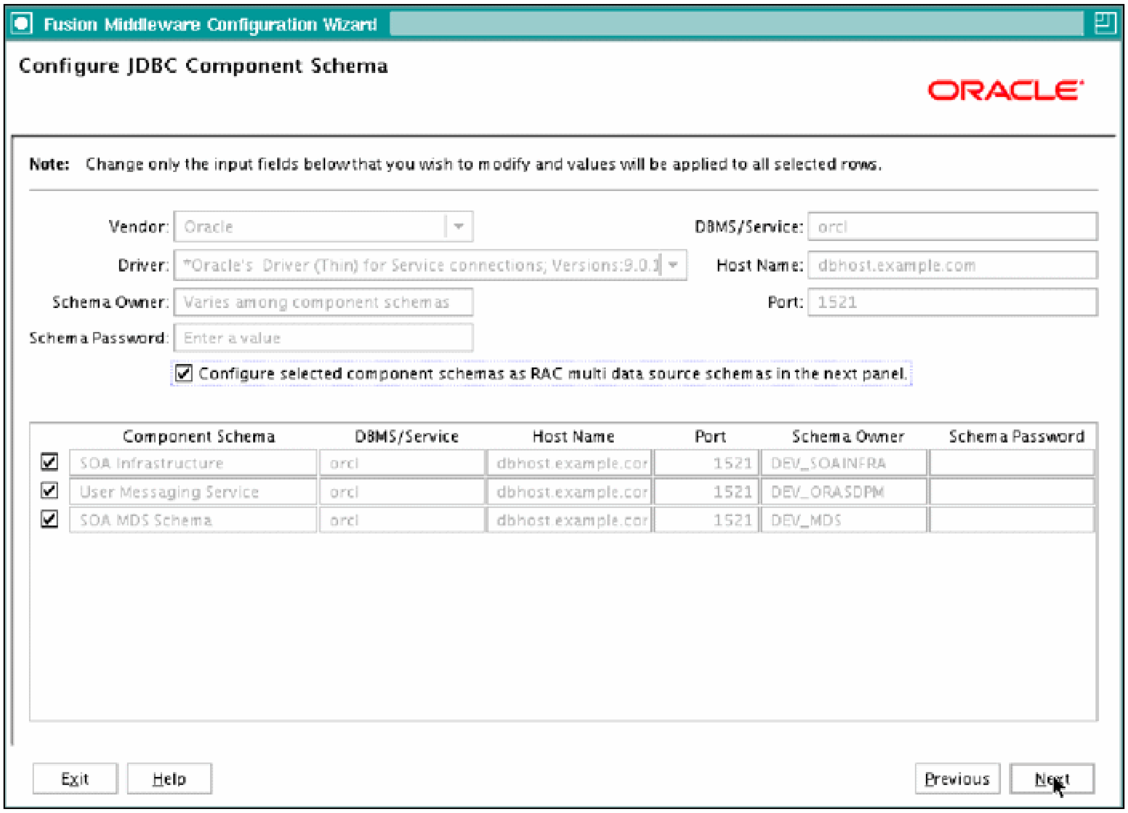
   Click **Next**.
6. If you get a "Conflict Detected" message that Oracle JRF is already defined in the domain, select the **Keep Existing Component** option and click **OK**.
7. In the Configure JDBC Component Schema screen (Figure 5–1), do the following:

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

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

   c. Click **Next**.

Figure 5–1 Configure JDBC Component Schema Screen



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

☑ 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	
☑	SOA MDS Schema	orcl	dbhost.example.cor	1521	DEV_MDS	

8. In the Configure RAC Multi Data Source Component Schema screen (Figure 5–2), do the following:

a. Select **SOA Infrastructure**.

b. Enter values for the following fields, specifying the connect information for the RAC database that was seeded with RCU.

– **Driver:** Select **Oracle driver (Thin) for RAC Service-Instance connections, Versions:10, 11**.

– **Service Name:** Enter the service name of the database; for example, `soaedg.mycompany.com`.

– **Username:** Enter the complete user name (including prefix) for the schemas. The user names shown in Figure 5–2 assume that `soedg` was used as prefix for schema creation from RCU.

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

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

d. Repeat for each RAC instance.

e. Deselect **SOA Infrastructure**.

f. Select **User Messaging Service**.

g. Repeat steps b, c, and d for the User Messaging Schema.

h. Deselect **User Messaging Service**.

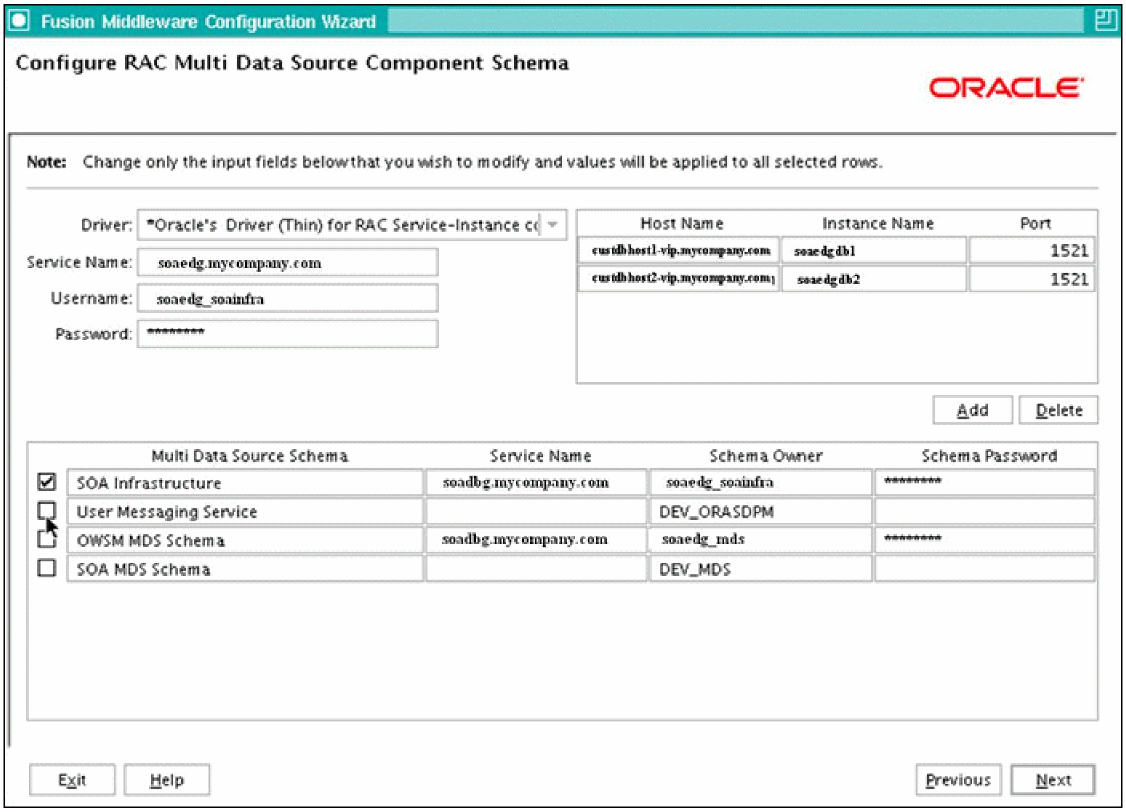
i. Select **SOA MDS Schema**.

j. Repeat steps b, c, and d for the SOA MDS Schema.

k. Leave the OWSM MDS Schema information as it is.

l. Click **Next**

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



Configure RAC Multi Data Source Component Schema

Note: Change only the input fields belowthat 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: soaedg.mycompany.com

Username: soaedg_soainfra

Password: *********

Host Name	Instance Name	Port
custdbhost1-vip.mycompany.com	soaedgdb1	1521
custdbhost2-vip.mycompany.com;	soaedgdb2	1521

	Multi Data Source Schema	Service Name	Schema Owner	Schema Password
☑	SOA Infrastructure	soadbg.mycompany.com	soaedg_soainfra	*********
□	User Messaging Service		DEV_ORASDPM	
□	OWSM MDS Schema	soadbg.mycompany.com	soaedg_mds	*********
□	SOA MDS Schema		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 Select Optional Configuration screen, select the following:

    - JMS Distributed Destinations
    - Managed Servers, Clusters, and Machines
    - Deployments and Services

    Click **Next**.

11. In the Select JMS Distributed Destination Type screen:

    - Select **UDD** from the drop down list for UMSJMSystemResource.
    - Select **UDD** from the drop down list for SOAJMSModule.

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 give it the attributes listed in Table 5–1. Then, add a new server called WLS_SOA2. The WLS_WSM1 and WLS_WSM2 managed servers should already be present because they are part of the domain that you are extending. In the end, the list of managed servers should match that in Table 5–1.

Table 5–1 Managed Servers

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled
WLS_SOA1	SOAHOST1VHN1	8001	n/a	No
WLS_SOA2	SOAHOST2VHN1	8001	n/a	No
WLS_WSM1	SOAHOST1	7010	n/a	No
WLS_WSM2	SOAHOST2	7010	n/a	No

Click **Next**.

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

Table 5–2 Clusters

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

Click **Next**.

14. In the Assign Servers to Clusters screen, assign servers to clusters as follows:
    - **SOA_Cluster:**
        - WLS_SOA1
        - WLS_SOA2
    - **WSM-PM_Cluster:**
        - WLS_WSM1
        - WLS_WSM2

    Click **Next**.

15. In the Configure Machines screen, do the following:
    - Delete the **LocalMachine** that appears by default.
    - Click the **Unix Machine** tab. The following entries appear (listed in Table 5–3):

Table 5–3 Machines

Name	Node Manager Listen Address
SOAHOST1	SOAHOST1
SOAHOST2	SOAHOST2
ADMINHOST	localhost

Leave all other fields to their default values.

Click **Next**.

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

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

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.)
    - The **oracle.sdp.***, and **oracle.soa.*** libraries should be targeted only to **SOA_Cluster**.
    - The **oracle.rules.*** library should be targeted only to **Admin Server** and **SOA_Cluster**.
    - The **wsm-pm** application should be targeted only to **WSM-PM_Cluster**.
    - The **oracle.wsm.seedpolicies** library should be targeted only to **WSM-PM_Cluster**.

      Target this library to the **SOA_Cluster** also only if you are planning to deploy WebLogic WebServices to it.

    Click **Next**.
18. In the Target Services to Clusters or Servers screen, ensure the following targets:
    - Target **JOC Startup Class** and **JOC Shutdown Class** only to **WSM-PM_Cluster**.
    - Target **mds-owsm**, **mds-owsm-rac0**, and **mds-owsm-rac1** to both **WSM-PM_Cluster** and **AdminServer**.

    Click **Next**.
19. In the Configuration Summary screen click **Extend**.

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

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

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

5.3 Restarting the Administration Server

Restart the Administration Server using the procedure in Section 4.7, "Starting the Administration Server on SOAHOST1."

5.4 Configuring Oracle Coherence for Deploying Composites

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

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

Enabling Communication for Deployment Using Unicast Communication

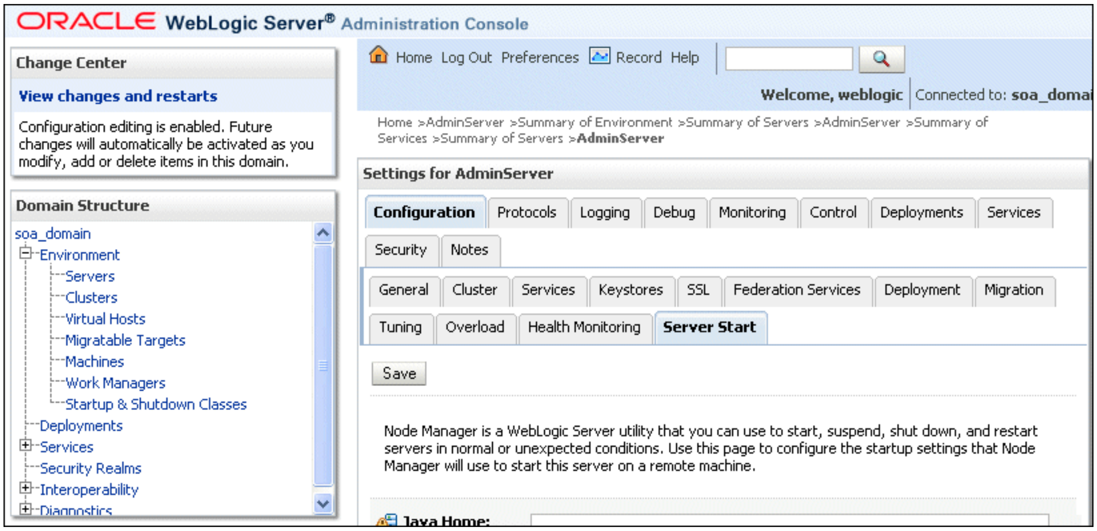
Multicast communication enables Oracle Fusion Middleware SOA to discover all of the members of a cluster to which to it deploys composites dynamically. However, unicast communication does not enable nodes to discover other cluster members in this way. Consequently, you must specify the nodes that belong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

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

Specify the nodes using the `tangosol.coherence.wka`*<n>* system property, where *<n>* is a number between 1 and 9. You can specify up to 9 nodes. Start the numbering at 1. This numbering must be sequential and must not contain gaps. In addition, specify the host name used by Oracle Coherence to create a cluster through the `tangosol.coherence.localhost` system property. This local host name should be the virtual host name used by the SOA server as the listener addresses (SOAHOST1VHN1 and SOAHOST2VHN1). Set this property by adding the `-Dtangosol.coherence.localhost` parameters to the Arguments field of the Oracle WebLogic Server Administration Console's Server Start tab (Figure 5–3).

Note: 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 5–3 Setting the Host Name Using the 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

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

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

   For WLS_SOA1, enter the following:

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

   For WLS_SOA2, enter the following:

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

Note: The Coherence cluster used for deployment uses port 8088 by default. This port can be changed by specifying a different port (for example, 8089) with 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.localport=8089
-Dtangosol.coherence.wka1.port=8089
-Dtangosol.coherence.wka2.port=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.localport=8089
-Dtangosol.coherence.wka1.port=8089
-Dtangosol.coherence.wka2.port=8089
```

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

Note: There should be no breaks in lines between the different `-D` parameters. Do not copy or paste the text from above to your Administration Console's arguments text field. This may result in HTML tags being inserted in the Java arguments. The text should not contain other text characters than those included the example above.

---

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.

5.5 Setting Connection Destination Identifiers for B2B Queues

Oracle B2B uses specific JMS Destination Member calls, and requires setting the Create Destination Identifier (CDI) for these calls to succeed. To set up the CDI:

1. Log into the Oracle WebLogic Server Administration Console.
2. In the Domain Structure window, expand the **Services** node, and then the **Messaging** node.
3. Click **JMS Modules**, and then **SOAJMSModule**.
4. Click **Lock and Edit**.
5. Click the **dist_B2BEventQueue_auto**, **Configuration**, and the **General** tab, and then click **Advanced**.
6. In the **Create Destination Identifier** field, add the following jndi name for the queue:

   `jms/b2b/B2BEventQueue`
7. Repeat these steps, creating the following Create Destination Identifiers for the queues listed below:
   - **dist_B2B_OUT_QUEUE_auto** : `jms/b2b/B2B_OUT_QUEUE`
   - **dist_B2B_IN_QUEUE_auto** : `jms/b2b/B2B_IN_QUEUE`
   - **dist_B2BBroadcastTopic_auto** : `jms/b2b/B2BBroadcastTopic`
   - **dist_XmlSchemaChangeNotificationTopic_auto** : `jms/fabric/XmlSchemaChangeNotificationTopic`
8. Click **Save and Active Changes**.

5.6 Disabling Host Name Verification for the WLS_SOA*n* Managed Server

This step is required if you have not set up the appropriate certificates to authenticate the different nodes with the Administration Server (see Chapter 8, "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 Enterprise Deployment topology configuration is complete as described in Chapter 8, "Setting Up Node Manager."

To disable host name verification, complete these steps:

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

5.7 Restarting the Node Manager on SOAHOST1

To restart the Node Manager on SOAHOST1:

1. Stop Node Manager by stopping the process associated with it:

   a. If it is running in the foreground in a shell, simply use **CTRL+C**.

   b. 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
   ```

5.8 Propagating the Domain Changes to the Managed Server Domain Directory

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

1. Create a copy of the managed server domain directory and the managed server applications directory.

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

   ```
   SOAHOST1> cd ORACLE_COMMON_HOME/common/bin

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

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

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

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

5.9 Starting the WLS_SOA1 Managed Server on SOAHOST1

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

1. Access the Administration Console at `http://ADMINVHN:7001/console`.

2. Click **Servers**.

3. Open the **Control** tab.
4. Select **WLS_SOA1**.
5. Click **Start**.

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

5.10 Validating the WLS_SOA1 Managed Server

To validate the WLS_SOA1 managed server, complete these steps:

1. Verify that the server status is reported as **Running** in the Admin Console. If the server is shown as **Starting** or **Resuming**, wait for the server status to change to **Started**. If another status is reported (such as **Admin** or **Failed**), check the server output log files for errors. See Section 10.7, "Troubleshooting" for possible causes.
2. Access `http://SOAHOST1VHN1:8001/soa-infra/` to verify status of WLS_SOA1.
3. Access `http://SOAHOST1VHN1:8001/b2bconsole/` to verify status of B2B.
4. Access `http://SOAHOST1VHN1:8001/integration/worklistapp/` to verify status of the worklist application. Before verifying access is granted, ensure that the WLS_WSM1 managed server is up and running.

> **Note:** Notice that, although the WLS_SOA1 server may be up, some applications may be in a failed state. Therefore, Oracle recommends verifying the URLs above and watch for errors pertaining each individual application in the server's output file.

5.11 Propagating the Domain Configuration to SOAHOST2 Using the unpack Utility

To propagate the domain configuration, complete these steps:

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

```
SOAHOST1> cd ORACLE_HOME/common/bin

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

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

```
SOAHOST2> cd ORACLE_HOME/common/bin

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

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

5.12 Extracting the XEngine Files in SOAHOST2

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

```
SOAHOST2>cd ORACLE_HOME/soa/thirdparty/edifecs
SOAHOST2>tar -xzvf XEngine.tar.gz
```

5.13 Restarting Node Manager on SOAHOST2

Perform the steps in Section 5.7, "Restarting the Node Manager on SOAHOST1" on SOAHOST2.

5.14 Starting and Validating the WLS_SOA2 Managed Server

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

1. Start the WLS_SOA2 managed server using the Administration Console.
2. Verify that the server status is reported as "Running" in the Admin Console. If the server is shown as "Starting" or "Resuming," wait for the server status to change to "Started." If another status is reported (such as "Admin" or "Failed"), check the server output log files for errors. See Section 10.7, "Troubleshooting" for possible causes.
3. Access `http://SOAHOST2VHN1:8001/soa-infra`.
4. Access `http://SOAHOST2VHN1:8001/b2bconsole` to verify status of B2B.
5. Access `http://SOAHOST2VHN1:8001/integration/worklistapp/` to verify status of the worklist application. Before verifying access is granted, ensure that at least one of the managed servers (WLS_WSM1 or WLS_WSM2) is up and running.

> **Note:** Although the WLS_SOA1 server may be up, some applications may be in a failed state. Therefore, Oracle recommends verifying the URLs above and watch for errors pertaining each individual application in the server's output file.

5.15 Configuring Oracle HTTP Server for the WLS_SOA*n* Managed Servers

To enable Oracle HTTP Server to route to the SOA_Cluster, which contains 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:

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

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

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

# B2B
<Location /b2bconsole>
    SetHandler weblogic-handler
    WebLogicCluster SOAHOST1VHN1:8001,SOAHOST2VHN1:8001
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>

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

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

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

#Required if attachments are added for workflow tasks
 <Location /ADFAttachmentHelper>
```

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

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

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

2. Restart Oracle HTTP Server on WEBHOST1 and WEBHOST2:

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

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

Sample scenarios:

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

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

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

5.16 Validating Access Through Oracle HTTP Server

Verify that the server status is reported as "Running" in the Admin Console. If the server is shown as "Starting" or "Resuming," wait for the server status to change to "Started." If another status is reported (such as "Admin" or "Failed"), check the server output log files for errors. See Section 10.7, "Troubleshooting" for possible causes.

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

- `http://webhostN:7777/soa-infra`
- `http://webhostN:7777/integration/worklistapp`
- `http://webhostN:7777/b2bconsole`
- `http://webhostN:7777/sdpmessaging/userprefs-ui`
- `http://webhostN:7777/soa/composer`

Validate SOA_Cluster through both Oracle HTTP Server instances.

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

5.17 Setting the Frontend HTTP Host and Port

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

1. In the WebLogic Server Administration Console, in the Change Center section, click **Lock & Edit**.
2. In the left pane, choose **Environment** in the Domain Structure window and then choose **Clusters**. The Summary of Clusters page appears.
3. Select the **SOA_Cluster** cluster.
4. Select **HTTP**.
5. Set the values for the following:
   - **Frontend Host**: soa.mycompany.com
   - **Frontend HTTPS Port**: 443
   - **Frontend HTTP Port**: 80
6. Click **Save**.
7. To activate the changes, click **Activate Changes** in the Change Center section of the Administration Console.
8. Restart the servers to make the Frontend Host directive in the cluster effective.

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 webservices fucntionality in Oracle Enterprise Manger 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).

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

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

Callback URL

The SOA system calculates the callback URL as follows:

- If a request to SOA originates from an external or internal service, then SOA uses the callback URL specified by the client.
- If a request to an external or internal asynchronous service originates from SOA, the callback URL is determined using the following method, in decreasing order of preference:

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

3. Use the callback URL as the frontend host specified in WLS for the SOA_Cluster. In this case, too, only one address can be specified and the recommendation is same as the one for *soa-infra-config.xml*.

4. Use the local host name as provided by WLS MBean APIs. This is not recommended in HA environments such as Enterprise Deployment.

5.18 Setting the WLS Cluster address for Direct Binding/RMI invocations to composites

When using direct binding composites, you must set the WLS Cluster address for the SOA_Cluster. To set the WLS Cluster address:

1. In the WebLogic Server Administration Console, in the **Change Center** section, click **Lock & Edit**.

2. In the left pane, choose **Environment** from the **Domain Structure** window, and then choose **Clusters**. The Summary of Clusters page appears.

3. Select the **SOA_Cluster** cluster.

4. In the **Configuration**, **General** tab, enter the following in the **Cluster Address** field:

   ```
   SOAHOST1VHN1:8001,SOAHOST2VHN1:8001
   ```

5. Click **Save**.

6. To activate the changes, click **Activate Changes** in the **Change Center** section of the Administration Console.

7. Restart the servers for the Frontend Host directive to take effect in the cluster.

> **Note:** For asynch request/response interactions over direct binding, the SOA composites must provide their jndi provider URL for the invoked service to look up the beans for callback.
>
> If soa-infra config properties are not specified, but the WebLogic Server Cluster address is specified, the cluster address from the JNDI provider URL is used. This cluster address can be a single DNS name which maps to the clustered servers' IP addresses or a comma separated list of server ip:port. Alternatively, the soa-infra config property `JndiProviderURL/SecureJndiProviderURL` can be used for the same purpose if explicitly set by users.

5.19 Configuring a Shared JMS Persistence Store

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

1. Log into the Oracle WebLogic Server Administration Console.

2. In the Domain Structure window, expand the **Services** node and then click the **Persistence Stores** node. The Summary of Persistence Stores page appears.

3. Select the persistence store (represented as a hyperlink) from the Name column of the table. The Settings page for the persistence store appear.

4. In the Configuration tab, enter the location on a persistent storage solution (such as NAS or SAN) that is available to other servers in the cluster in the Directory field. Specifying this location enables pending JMS messages to be sent. The location should follow the following directory structure:

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

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

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

5.20 Configuring a Default Persistence Store for Transaction Recovery

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

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

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

1. Log into the Oracle WebLogic Server Administration Console.
2. In the Domain Structure window, expand the **Environment** node and then click the **Servers** node. The Summary of Servers page appears.
3. Click the name of the server (represented as a hyperlink) in Name column of the table. The settings page for the selected server appears and defaults to the Configuration tab.
4. Click the **Services** tab.
5. In the Default Store section of the page, enter the path to the folder where the default persistent stores will store its data files. The directory structure of the path is as follows:

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

6. Click **Save**.

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

5.21 Enabling High Availability for Oracle File and FTP Adapters

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

> **Note:** The operation described above is necessary only if your application requires these adapters.

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

5.21.1 Using the Database Mutex Locking Operation

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

> **Note:** The steps and configuration options for the FTP adapter are exactly the same as the options for the file adapter. The connection factory to be used for FTP HA configuration is `eis/Ftp/HAFtpAdapter` which appears under the Outbound Connection Pools for the FTPAdapter deployment.

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

1. **Create Database Tables**

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

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

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

   a. Log into your Oracle WebLogic Server console. To access the console navigate to `http://`*`servername`*:*`portnumber`*`/console`.

   b. Click **Deployments** in the left pane for Domain Structure.

   c. Click **FileAdapter** under Summary of Deployments on the right pane.

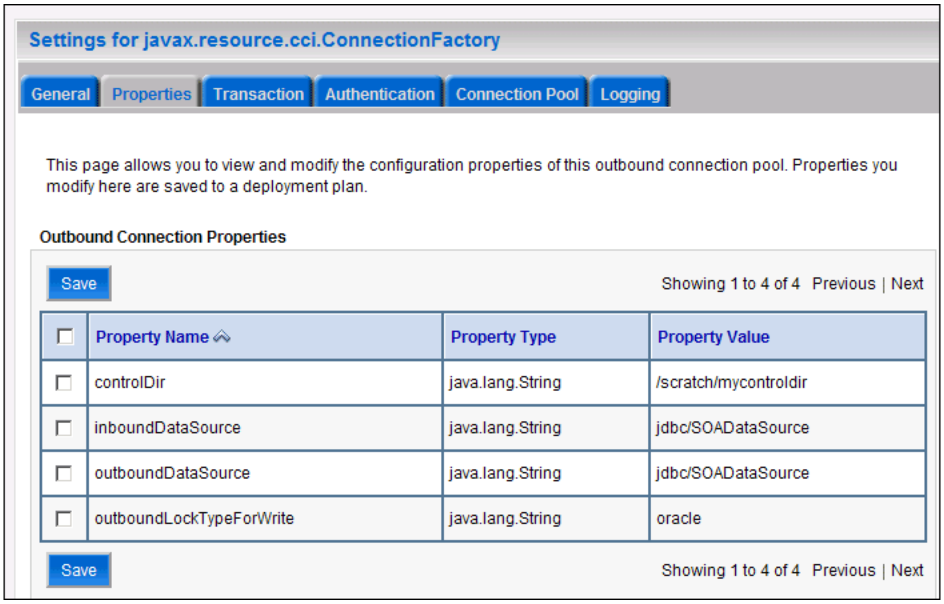
d. Click the **Configuration** tab.

e. Click the **Outbound Connection Pools** tab, and expand **javax.resource.cci.ConnectionFactory** to see the configured connection factories.

f. Click **eis/HAFileAdapter**. The Outbound Connection Properties for the connection factory corresponding to high availability is displayed.

g. The connection factory properties appear as shown in Figure 5–4.

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


Settings for javax.resource.cci.ConnectionFactory
General
Properties
Transaction
Authentication
Connection Pool
Logging
Save
Showing 1 to 4 of 4 Previous | Next

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

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

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

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

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

`inboundDataSource:` Set the value to `jdbc/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.

outboundDataSourceLocal: Set the value to jdbc/SOALocalTxDataSource. This is the datasource where the schemas corresponding to high availability are pre-created.

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

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

oracle: The adapter uses Oracle Database sequence.

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

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

h. Click **Save** after you update the properties. The Save Deployment Plan page appears.

i. Enter a shared storage location for the deployment plan. The directory structure is as follows:

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

j. Click **Save and Activate**.

k. Configure BPEL Process or Mediator Scenario to use the connection factory as shown in the following example (in the jca file included in the composite for the binding component):

```
<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.

5.22 Update the B2B Instance Identifier for transports

To set up File, FTP, or Email transports in a high availability environment, specify a unique name for each instance by using b2b.HAInstanceName *unique_instance_name*. If you use *ServerName* for the value, Oracle B2B retrieves the WebLogic Server name as the HAInstanceName. To specify a unique name for each instance:

1. Log in to Oracle Enterprise Manager Fusion Middleware Control with the username and password specified in Section 4.5, "Creating boot.properties for the Administration Server on SOAHOST1."
2. On the navigation tree on the left, expand **Farm_<domain_name>**, **SOA**, and then right click on the **soa-infra<server_name>**, and select the **SOA Administration**, and then **B2B Server Properties**.
3. Click on **More B2B Configuration Properties...** on the right.
4. Click the **b2b MBean**.
5. Click the **Operations** tab.
6. Click **addProperty** in the list on the right.
7. In the **Key** field enter **b2b.HAInstanceName**.
8. In the value field enter *ServerName*.
9. Click **Invoke**.

5.23 Scaling the Oracle Database Adapter

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 this release that approach has now been superseded. If you were using this approach previously, you can simply remove (in db.jca) or clear (Logical Delete Page of wizard) the MarkReservedValue, and you 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 a non-recoverable situation in a high availability environment is minimized.
- No unique `MarkReservedValue` must be specified. Previously, for this to work you would have 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*.

5.24 Backing Up the Installation

After you have verified that the extended domain is working, back up the installation. This is a quick backup for the express purpose of immediate restore in case of problems in the further steps. The backup destination is the local disk. This backup can be discarded once the enterprise deployment setup is complete. At this point, the regular deployment-specific backup and recovery process can be initiated. The *Oracle*

Fusion Middleware Administrator's Guide provides further details. For information on describing the Oracle HTTP Server data that must be backed up and restored, refer to the "Backup and Recovery Recommendations for Oracle HTTP Server" section in this guide. For information on how to recover components, see "Recovery of Components" and "Recovery After Loss of Component" sections in the guide. For recommendations specific to recovering from the loss of a host, see the "Recovering Oracle HTTP Server to a Different Host" in the guide. Also refer to the *Oracle Database Backup and Recovery Guide* for information on database backup.

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

1. Back up the web tier:

   a. Shut down the instance using `opmnctl`.

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

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

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

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

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

   d. Start the instance using `opmnctl`:

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

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

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

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

> **Note:** ORACLE_HOME should be backed up if any changes are made to the XEngine configuration that are part of your B2B setup. These files are located under `ORACLE_HOME/soa/thirdparty/edifecs/XEngine`. To back up ORACLE_HOME, execute the following command:
>
> ```
> SOAHOST1> tar -cvpf fmwhomeback.tar MW_HOME
> ```

6 Extending the Domain to Include Oracle BPM

The Oracle BPM system is created using the WL_HOME and ORACLE_HOME installed in previous chapters on a shared storage. Oracle BPM requires these WL_HOME and ORACLE_HOME homes to be patched to the Oracle FMW 11.1.1.4 (PS3) patchset level before the WebLogic Configuration Wizard steps are performed to extend a domain.

You can enable BPM in a Fusion Middleware installation in the following two ways:

- Extend an existing domain that contains an Administration Server (and optionally other non-SOA servers) to include SOA and BPM (in one single Configuration Wizard session). For this approach, see Section 6.1, "Option 1: Extending a Domain to Include SOA and BPM."
- Extend a domain that already contains SOA (and optionally other non-SOA servers) to BPM. For this approach, see Section 6.2, "Option 2: Extending a SOA Domain to Include Oracle BPM.".

6.1 Option 1: Extending a Domain to Include SOA and BPM

This section describes how to extend a domain with SOA and BPM components using the Configuration Wizard. You can extend the resulting domain to add BAM. It is assumed that a SOA ORACLE_HOME (binaries) has already been installed, patched to latest patch set if applicable, and is available from SOAHOST1 and SOAHOST2. it is also assumed that a domain with an Administration Server has been created. This is the domain that is extended in this chapter to support SOA components.

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

This section contains the following topics:

- Section 6.1.1, "Backing Up the Existing Installation"
- Section 6.1.2, "Enabling VIP2 on SOAHOST1 and VIP3 on SOAHOST2"
- Section 6.1.3, "Running the Configuration Wizard on SOAHOST1 to Extend the Current Domain"
- Section 6.1.4, "Configuring Oracle Coherence for Deploying Composites"
- Section 6.1.5, "Setting Connection Destination Identifiers for B2B Queues"
- Section 6.1.6, "Disabling Host Name Verification for the WLS_SOA1, WLS_SOA2, Managed Servers"

- Section 6.1.7, "Propagating the Domain Changes to the Managed Server Domain Directory"
- Section 6.1.8, "Starting and Validating the WLS_SOA1 Managed Server"
- Section 6.1.9, "Propagating the Domain Configuration to SOAHOST2 Using the Unpack Utility"
- Section 6.1.10, "Extracting the XEngine Files in SOAHOST2"
- Section 6.1.11, "Starting and Validating the WLS_SOA2 Managed Server"
- Section 6.1.12, "Configuring Oracle HTTP Server for WLS_SOAn Managed Servers"
- Section 6.1.13, "Validating Access Through Oracle HTTP Server"
- Section 6.1.14, "Setting the Frontend HTTP Host and Port"
- Section 6.1.15, "Setting the WLS Cluster address for Direct Binding/RMI invocations to composites"
- Section 6.1.16, "Configuring a Shared JMS Persistence Store"
- Section 6.1.17, "Configuring a Default Persistence Store for Transaction Recovery"
- Section 6.1.18, "Enabling High Availability for Oracle File and FTP Adapters"
- Section 6.1.19, "Scaling the Oracle Database Adapter"
- Section 6.1.20, "Backing Up the Installation"

6.1.1 Backing Up the Existing Installation

Before beginning the procedures in this chapter, Oracle recommends backing 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
```

These commands create a backup of the installation files for both Oracle WebLogic Server and Oracle Fusion Middleware, as well as the domain configuration.

6.1.2 Enabling VIP2 on SOAHOST1 and VIP3 on SOAHOST2

The procedures in this section are required for server migration of WLS_SOA1 and WLS_SOA2.

Associate the WLS_SOA1 Server and WLS_SOA2 with virtual hostnames (SOAHOST1VHN1 and SOAHOST2VHN1). Check that these virtual hostnames 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 on Linux, run the `ifconfig` command as root:

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

For example:

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

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

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

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

```
/bin/ping 100.200.140.206
```

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

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

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

1. Ensure that the database where you installed the repository is running. For Oracle RAC databases, Oracle recommends all instances remain running, so that the validation check later in the process is more reliable.
2. Change the directory to the location of the Configuration Wizard. This is within the SOA home directory.

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

3. Start the Oracle Fusion Middleware 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 products:
   - Select the following products:
     - Oracle BPM Suite - 11.1.1.0 [soa]
     - Oracle SOA Suite - 11.1.1.0 [soa] (this should be selected automatically when selecting Oracle BPM Suite)
     - Oracle WSM Policy Manager 11.1.1.0 [oracle_common] (this should be selected automatically when selecting Oracle BPM Suite)
     - Oracle Enterprise Manager - 11.1.1.0 [oracle_common]
     - Oracle JRF - 11.1.1.0 [oracle_common] (this should be selected automatically and grayed out)

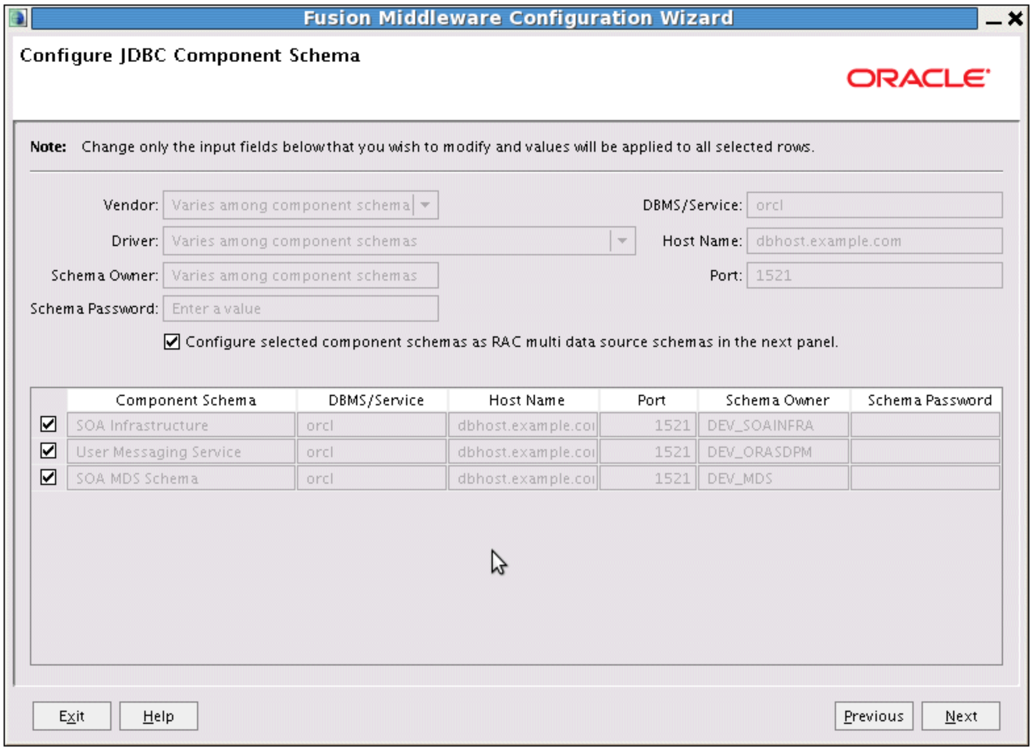
     If you accidentally deselect some of the targets, make sure the following are selected:

     - Oracle SOA
     - Oracle BPM Suite

     Click **Next**.
7. In the Configure JDBC Component Schema screen, do the following:
   - Select **SOA Infrastructure**, **User Messaging Service**, and **SOA MDS Schema**.

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

Figure 6–1 Configure JDBC Component Schema Screen



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: Varies among component schema

Driver: Varies among component schemas

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.coı	1521	DEV_SOAINFRA	
☑	User Messaging Service	orcl	dbhost.example.coı	1521	DEV_ORASDPM	
☑	SOA MDS Schema	orcl	dbhost.example.coı	1521	DEV_MDS	

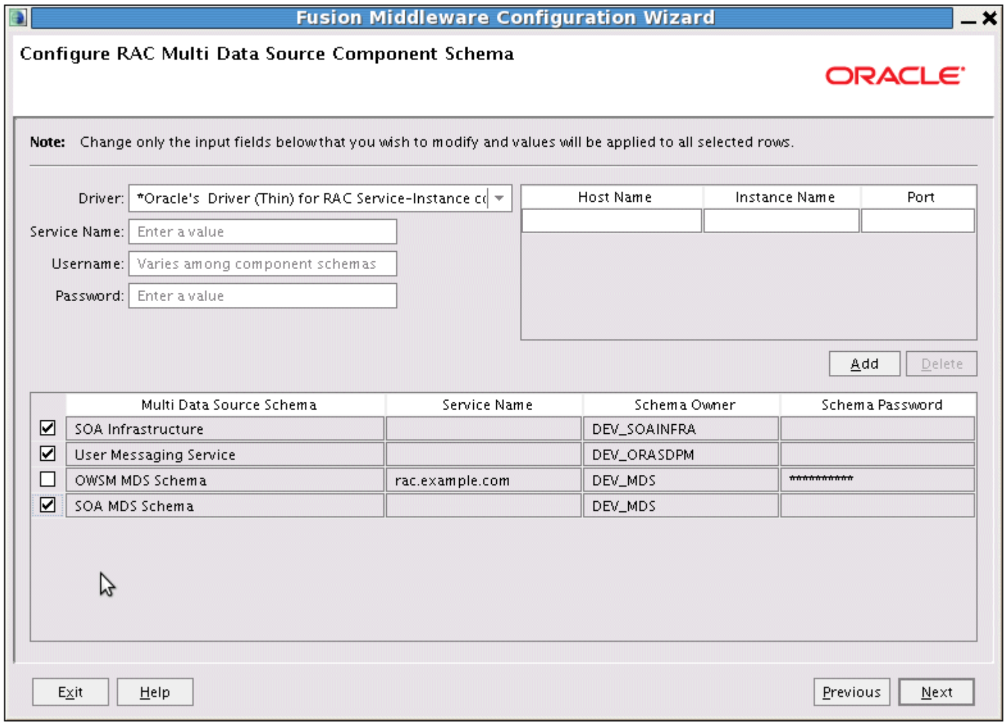
Configure JDBC Component Schema screen

****

Click **Next**.

8. The is displayed. 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):

Figure 6–2 Configure RAC Multi Data Source Component Schema Screen



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: Enter a value

Username: Varies among component schemas

Password: Enter a value

Host Name	Instance Name	Port

	Multi Data Source Schema	Service Name	Schema Owner	Schema Password
☑	SOA Infrastructure		DEV_SOAINFRA	
☑	User Messaging Service		DEV_ORASDPM	
□	OWSM MDS Schema	rac.example.com	DEV_MDS	************
☑	SOA MDS Schema		DEV_MDS	

In this screen, do the following:

You can select all schemas and enter values that are common to all. Leave the OWSM MDS Schema information as it is, since it was already configured in Chapter 4.

Enter values for the following fields, specifying the connect information for the Oracle RAC database that was seeded with RCU. Enter this information for each schema:

a. Enter the host name, instance name, and port for the first Oracle RAC DB instance.

b. Click **Add**.

c. Repeat for each Oracle RAC instance.

d. Driver: Select **Oracle driver (Thin) for RAC Service-Instance connections, Versions:10, 11**.

e. Service Name: Enter the service name of the database, for example, **soaedg.mycompany.com**

f. 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

g. Password: Enter the password to use to access the schemas.

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 Destinations
    - Managed Servers, Clusters, and Machines
    - Deployments and Services

    Click **Next**.

11. In the Select JMS Distributed Destination Type screen, Select UDD from the drop-down list for all Fusion Middleware Components' JMS Modules.

    > **Note:** Oracle does not support using WDDs for Fusion Middleware components

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

    A server named `soa_server1` is created automatically. Rename this to WLS_SOA1 and add a new server named WLS_SOA2. Give these servers the attributes listed in Table 6–1. Do not modify the other servers that appear 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 cluster listed in Table 6–2. Do not modify the other clusters that display in this screen; leave them as they are.

Table 6–2 Cluster

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 follows:

    - **SOA_Cluster:**
      - WLS_SOA1
      - WLS_SOA2

    Click **Next**.

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

    - Click **Delete** to remove the default **LocalMachine**.
    - Click the **Unix Machine** tab. SOAHOST1 and SOAHOST2 machines appear with the following entries Table 6–3):

Table 6–3 Machines

Name	Node Manager Listen Address
SOAHOST1	SOAHOST1
SOAHOST2	SOAHOST2
ADMINHOST	localhost

Leave all other fields to their default values.

Click **Next**.

16. In the Assign Servers to Machines screen, do the following:

    - Assign AdminServer to ADMINHOST.
    - Assign WLS_SOA1 to SOAHOST1.
    - Assign WLS_SOA2 to SOAHOST2.
    - Assign WLS_WSM1 to SOAHOST1.
    - Assign WLS_WSM2 to SOAHOST2.

    Click **Next**.

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

    - Target **WSM-PM** only to **WSM-PM_Cluster**.
    - Target the **oracle.sdp.***, **oracle.bpm.***, and **oracle.soa.*** deployments only to **SOA_Cluster**.
    - The **oracle.rules.*** library should be targeted only to **Admin Server** and **SOA_Cluster**.
    - Target the **oracle.wsm.seedpolicies** library only to **WSM-PM_Cluster** and **SOA_Cluster** (if the SOA_Cluster is going to contain any webservices itself).

    Click **Next**.

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

    - Target **JOC Startup Class** and **JOC Shutdown Class** only to **WSM-PM_Cluster**.
    - Target the **mds-owsm**, **mdw-owsm-rac0** and **mds-owsm-rac1** datasources to the **WSM-PM_Cluster** and the **AdminServer**

    Click **Next**.

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

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

    You must restart the Administration Server for this configuration to take effect. to restart the Administration Server, use the procedure in Section 4.7, "Starting the Administration Server on SOAHOST1."

6.1.4 Configuring Oracle Coherence for Deploying Composites

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

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

Enabling Communication for Deployment Using Unicast Communication

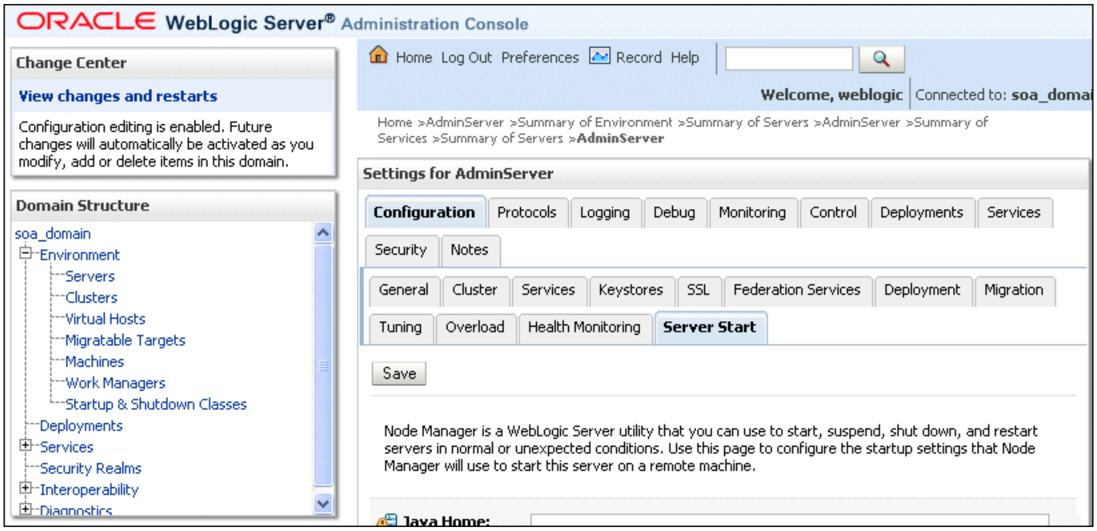
Multicast communication enables Oracle Fusion Middleware SOA to discover all of the members of a cluster to which to it deploys composites dynamically. However, unicast communication does not enable nodes to discover other cluster members in this way. Consequently, you must specify the nodes that belong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

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

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

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

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

   For WLS_SOA1, enter the following:

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

   For WLS_SOA2, enter the following:

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

   > **Note:** There should be no breaks in lines between the different `-D` parameters. Do not copy or paste the text from above to your Administration Console's arguments text field. This may result in HTML tags being inserted in the Java arguments. The text should not contain other text characters than those included the example above.

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.

6.1.5 Setting Connection Destination Identifiers for B2B Queues

Oracle B2B uses specific JMS Destination Member calls, and requires setting the Create Destination Identifier (CDI) for these calls to succeed. To set up the CDI:

1. Log into the Oracle WebLogic Server Administration Console.
2. In the Domain Structure window, expand the **Services** node, and then the **Messaging** node.
3. Click **JMS Modules**, and then **SOAJMSModule**.
4. Click **Lock and Edit**.
5. Click the **dist_B2BEventQueue_auto**, **Configuration**, and the **General** tab, and then click **Advanced**.
6. In the **Create Destination Identifier** field, add the following jndi name for the queue:

   ```
   jms/b2b/B2BEventQueue
   ```

7. Repeat these steps, creating the following Create Destination Identifiers for the queues listed below:
   - **B2B_OUT_QUEUE**: `jms/b2b/B2B_IN_QUEUE`
   - **B2B_IN_QUEUE**: `jms/b2b/B2B_OUT_QUEUE`
   - **B2BBroadcastTopic**: `jms/b2b/B2BBroadcastTopic`
   - **XmlSchemaChangeNotificationTopic**: `jms/fabric/XmlSchemaChangeNotificationTopic`
8. Click **Save and Active Changes**.

6.1.6 Disabling Host Name Verification for the WLS_SOA1, WLS_SOA2, 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 8, "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 Enterprise Deployment topology configuration is complete as described in Chapter 8, "Setting Up Node Manager."

To disable host name verification:

1. Log in to Oracle WebLogic Server Administration Console.
2. Click **Lock and Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**.

   The Summary of Servers page appears.
5. Select **WLS_SOA1** (represented as a hyperlink) from the **Names** column of the table.

   The Settings page appears.
6. Select the **SSL** tab.
7. Expand the **Advanced** section of the page.
8. Set Hostname Verification to **None**.
9. Click **Save**.
10. Repeat Steps 4 through 8 for the WLS_SOA2 server.
11. Save and activate the changes.
12. This change requires a restart of the Administration Server and Node Managers. To restart the Admin Server, follow the steps provided in section Section 4.7, "Starting the Administration Server on SOAHOST1."

    a. To restart the Admin Server follow the steps provided in Section 5.3, "Restarting the Administration Server.".

    b. To restart Node Manager on SOAHOST1, use the procedure described in Section 5.7, "Restarting the Node Manager on SOAHOST1."

       Repeat for Node Manager in SOAHOST2

6.1.7 Propagating the Domain Changes to the Managed Server Domain Directory

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

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

```
SOAHOST1> cd ORACLE_COMMON_HOME/common/bin

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

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

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

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

6.1.8 Starting and Validating the WLS_SOA1 Managed Server

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:

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

   b. Choose **Servers**.

      The Summary of Servers screen appears.

   c. Click the **Control** tab.

   d. Select **WLS_SOA1**, then click **Start**.

2. Verify that the server status is reported as **Running**. 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 Chapter 10.7, "Troubleshooting" for possible causes.

3. Access the following URLs:

   `http://SOAHOST1VHN1:8001/soa-infra/` to verify status of WLS_SOA1.

   `http://SOAHOST1VHN1:8001/soa/composer/` to verify status of SOA process composer.

   `http://SOAHOST1VHN1:8001/integration/worklistapp/` to verify status of the worklist application.

   `http://SOAHOST1VHN1:8001/b2bconsole/` to verify status of B2B.

   `http://SOAHOST1VHN1:8001/sdpmessaging/userprefs-ui/` to verify status of messaging system preferences

   `http://SOAHOST1VHN1:8001/bpm/composer/` and login to the composer application.

   `http://SOAHOST1VHN1:8001/bpm/workspace/` and login to the workspace application.

6.1.9 Propagating the Domain Configuration to SOAHOST2 Using the Unpack Utility

To propagate the domain configuration to SOAHOST2:

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

   ```
   SOAHOST1> cd ORACLE_COMMON_HOME/common/bin

   SOAHOST1> scp soadomaintemplateExtSOABPM.jar oracle@node2:ORACLE_COMMON_
   HOME/common/bin
   ```

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

```
SOAHOST2> cd ORACLE_COMMON_HOME/common/bin

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

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

6.1.10 Extracting the XEngine Files in SOAHOST2

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

```
SOAHOST2>cd ORACLE_HOME/soa/thirdparty/edifecs
SOAHOST2>tar -xzvf XEngine.tar.gz
```

6.1.11 Starting and Validating the WLS_SOA2 Managed Server

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 screen appears.
   c. Click the **Control** tab.
   d. Select **WLS_SOA2** and then click **Start**.
2. Verify that the server status is reported as **Running**. 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 Chapter 10.7, "Troubleshooting" for possible causes.
3. Access the following URLs:

   `http://SOAHOST2VHN1:8001/soa-infra` to verify status of WLS_SOA2.

   `http://SOAHOST2VHN1:8001/soa/composer` to verify status of soa process composer.

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

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

   `http://SOAHOST2VHN1:8001/b2bconsole` to verify status of B2B.

`http://SOAHOST2VHN1:8001/sdpmessaging/userprefs-ui` to verify status of messaging system preferences

`http://SOAHOST2VHN1:8001/bpm/composer` and login to the composer application.

`http://SOAHOST2VHN1:8001/bpm/workspace` and login to the workspace application.

6.1.12 Configuring Oracle HTTP Server for WLS_SOA*n* Managed Servers

To enable Oracle HTTP Server to route to the SOA_Cluster, which contains 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:

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

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

# B2B
<Location /b2bconsole>
    SetHandler weblogic-handler
    WebLogicCluster SOAHOST1VHN1:8001,SOAHOST2VHN1:8001
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>

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

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

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

#Required if attachments are added for workflow tasks
 <Location /ADFAttachmentHelper>
    SetHandler weblogic-handler
    WebLogicCluster SOAHOST1VHN1:8001,SOAHOST2VHN1:8001
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>

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

# BPM
<Location /bpm/composer>
    SetHandler weblogic-handler
    WebLogicCluster SOAHOST1VHN1:8001,SOAHOST2VHN1:8001
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>

# BPM
<Location /bpm/workspace>
    SetHandler weblogic-handler
    WebLogicCluster SOAHOST1VHN1:8001,SOAHOST2VHN1:8001
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>
```

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://soa.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 soa.mycompany.com:443, 7777, admin.mycompany:80, and you@youraddress that are noted in this document serve as examples only. Enter values based on the actual environment.

---

2. Restart Oracle HTTP Server on WEBHOST1 and WEBHOST2:

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

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

Some example scenarios:

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

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

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

6.1.13 Validating Access Through Oracle HTTP Server

Verify that the server status is reported as **Running**. 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 10.7, "Troubleshooting" for possible causes.

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

- `http://WEBHOST1:7777/soa-infra`
- `http://WEBHOST2:7777/soa-infra`
- `http://WEBHOST1:7777/soa/composer`
- `http://WEBHOST2:7777/soa/composer`
- `http://WEBHOST1:7777/integration/worklistapp`
- `http://WEBHOST2:7777/integration/worklistapp`
- `http://WEBHOST1:7777/sdpmessaging/userprefs-ui`
- `http://WEBHOST2:7777/sdpmessaging/userprefs-ui`

- `http://WEBHOST1:7777/b2bconsole`
- `http://WEBHOST2:7777/b2bconsole`
- `http://WEBHOST1:7777/bpm/composer`
- `http://WEBHOST2:7777/bpm/composer`
- `http://WEBHOST1:7777/bpm/workspace`
- `http://WEBHOST2:7777/bpm/workspace`

You can also verify these URLs using your load balancer address:

- `http://soa.mycompany.com:80/soa-infra`
- `http://soa.mycompany.com:80/soa/composer`
- `http://soa.mycompany.com:80/integration/worklistapp`
- `http://soa.mycompany.com:80/sdpmessaging/userprefs-ui`
- `http://soa.mycompany.com:80/b2bconsole`
- `http://soa.mycompany.com:80/bpm/composer`
- `http://soa.mycompany.com:80/bpm/workspace`

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

6.1.14 Setting the Frontend HTTP Host and Port

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

1. In the WebLogic Server Administration Console, in the Change Center section, click **Lock & Edit**.
2. In the left pane, choose **Environment** in the Domain Structure window and then choose **Clusters**. The Summary of Clusters page appears.
3. Select the **SOA_Cluster** cluster.
4. Select **HTTP**.
5. Set the values for the following:
   - **Frontend Host**: soa.mycompany.com
   - **Frontend HTTPS Port**: 443
   - **Frontend HTTP Port**: 80
6. Click **Save**.
7. To activate the changes, click **Activate Changes** in the **Change Center** section of the Administration Console.
8. Restart the servers to make the Frontend Host directive in the cluster effective.

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 webservices fucntionality in Oracle Enterprise Manger 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).

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

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

Callback URL

The SOA system calculates the callback URL as follows:

- If a request to SOA originates from an external or internal service, then SOA uses the callback URL specified by the client.
- If a request to an external or internal asynchronous service originates from SOA, the callback URL is determined using the following method, in decreasing order of preference:

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

3. Use the callback URL as the frontend host specified in WLS for the SOA_Cluster. In this case, too, only one address can be specified and the recommendation is same as the one for *soa-infra-config.xml*.

4. Use the local host name as provided by WLS MBean APIs. This is not recommended in HA environments such as Enterprise Deployment.

6.1.15 Setting the WLS Cluster address for Direct Binding/RMI invocations to composites

When using direct binding composites, you must set the WLS Cluster address for the SOA_Cluster. To set the WLS Cluster address:

1. In the WebLogic Server Administration Console, in the **Change Center** section, click **Lock & Edit**.

2. In the left pane, choose **Environment** from the **Domain Structure** window, and then choose **Clusters**. The Summary of Clusters page appears.

3. Select the **SOA_Cluster** cluster.

4. In the **Configuration**, **General** tab, enter the following in the **Cluster Address** field:

   ```
   SOAHOST1VHN1:8001,SOAHOST2VHN1:8001
   ```

5. Click **Save**.

6. To activate the changes, click **Activate Changes** in the **Change Center** section of the Administration Console.

7. Restart the servers for the Frontend Host directive to take effect in the cluster.

> **Note:** For asynch request/response interactions over direct binding, the SOA composites must provide their jndi provider URL for the invoked service to look up the beans for callback.
>
> If soa-infra config properties are not specified, but the WLS Cluster address is specified, the cluster address from the JNDI provider URL is used. This cluster address can be a single DNS name which maps to the clustered servers' IP addresses or a comma separated list of server ip:port. Alternatively, the soa-infra config property `JndiProviderURL/SecureJndiProviderURL` can be used for the same purpose if explicitly set by users.

6.1.16 Configuring a Shared JMS Persistence Store

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

1. Log into the Oracle WebLogic Server Administration Console.

2. In the Domain Structure window, expand the **Services** node and then click the **Persistence Stores** node.

   The Summary of Persistence Stores page appears.

3. Select the persistence store (represented as a hyperlink) from the **Name** column of the table.

The Settings page for the persistence store appear.

4. In the **Configuration** tab, enter the location on a persistent storage solution (such as NAS or SAN) that is available to other servers in the cluster in the Directory field.

   Specifying this location enables pending JMS messages to be sent. The location should follow the following directory structure:

   ```
   ORACLE_BASE/admin/domain_name/soa_cluster_name/jms
   ```

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

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

6.1.17 Configuring a Default Persistence Store for Transaction Recovery

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

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

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

1. Log into the Oracle WebLogic Server Administration Console.
2. In the Domain Structure window, expand the **Environment** node and then click the **Servers** node. The Summary of Servers page appears.
3. Click the name of the server (represented as a hyperlink) in Name column of the table. The settings page for the selected server appears and defaults to the Configuration tab.
4. Click the **Services** tab.
5. In the Default Store section of the page, enter the path to the folder where the default persistent stores will store its data files. The directory structure of the path is as follows:

   ```
   ORACLE_BASE/admin/domain_name/soa_cluster_name/tlogs
   ```

6. Click **Save**.

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

6.1.18 Enabling High Availability for Oracle File and FTP Adapters

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

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

6.1.18.1 Using the Database Mutex Locking Operation

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

> **Note:** The steps and configuration options for the FTP adapter are exactly the same as the options for the file adapter. The connection factory to be used for FTP HA configuration is `eis/Ftp/HAFtpAdapter` which appears under the Outbound Connection Pools for the FTPAdapter deployment.

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

1. **Create Database Tables**

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

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

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

   a. Log into your Oracle WebLogic Server console. To access the console navigate to `http://`*`servername`*:*`portnumber`*`/console`.

   b. Click **Deployments** in the left pane for Domain Structure.

   c. Click **FileAdapter** under Summary of Deployments on the right pane.

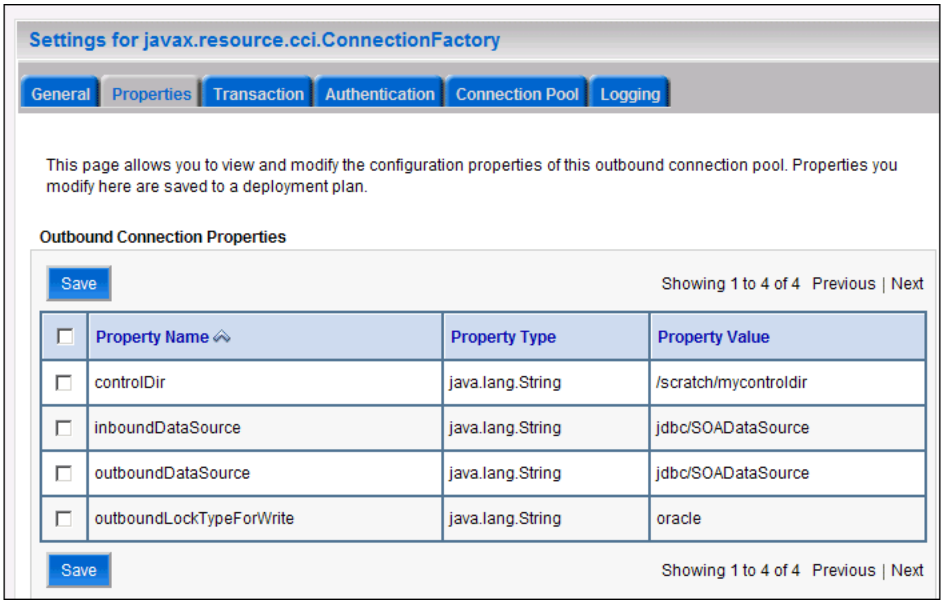
   d. Click the **Configuration** tab.

   e. Click the **Outbound Connection Pools** tab, and expand **javax.resource.cci.ConnectionFactory** to see the configured connection factories.

f. Click **eis/HAFileAdapter**. The Outbound Connection Properties for the connection factory corresponding to high availability is displayed.

g. The connection factory properties appear as shown in Figure 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

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

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

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

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

`inboundDataSource`: Set the value to `jdbc/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.

`outboundDataSourceLocal`: Set the value to `jdbc/SOADataSource`. This is the datasource where the schemas corresponding to high availability are pre-created.

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

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

oracle: The adapter uses Oracle Database sequence.

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

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

h. Click **Save** after you update the properties. The Save Deployment Plan page appears.

i. Enter a shared storage location for the deployment plan. The directory structure is as follows:

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

j. Click **Save and Activate**.

k. Configure BPEL Process or Mediator Scenario to use the connection factory as shown in the following example:

```
<adapter-config name="FlatStructureOut" adapter="File Adapter"
xmlns="http://platform.integration.oracle/blocks/adapter/fw/metadata">
  <connection-factory location="eis/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.

6.1.19 Scaling the Oracle Database Adapter

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 this release that approach has now been superseded. If you were using this approach previously, you can simply remove (in db.jca) or clear (Logical Delete Page of wizard) the MarkReservedValue, and you 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 a non-recoverable situation in a high availability environment is minimized.
- No unique `MarkReservedValue` must be specified. Previously, for this to work you would have 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*.

6.1.20 Backing Up the Installation

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

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

1. Back up the web tier:

   a. Shut down the instance using `opmnctl`.

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

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

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

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

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

   d. Start the instance using `opmnctl`:

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

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

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

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

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

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

6.2 Option 2: Extending a SOA Domain to Include Oracle BPM

In this step, you extend the domain created in Section 5, "Extending the Domain for SOA Components" to contain BPM. You can also extend the resulting domain to add BAM. It is assumed that SOA ORACLE_HOME (binaries) has already been installed, patched to the Oracle FMW 11.1.1.4 (PS3) patchset level, and is available from SOAHOST1 and SOAHOST2 (this is required before the WebLogic Configuration Wizard steps are performed to extend the domain).

The instructions in this section assume that Node Manager, Admin Server, SOA Servers and WSM Servers exist and have been configured as described in previous chapters to run a SOA system. Server migration, transaction logs, coherence, and all other configuration steps for the SOA System have already been performed and will be used by BPM. BPM is added as a superset of the existing configuration.

This section contains the following topics:

- Section 6.2.1, "Backing Up the Existing Installation"
- Section 6.2.2, "Running the Configuration Wizard on SOAHOST1 to Extend a SOA Domain to Include BPM"
- Section 6.2.3, "Configuring a JMS Persistence Store for BPM JMS"
- Section 6.2.4, "Propagating the Domain Configuration to the managed server directory in SOAHOST1 and to SOAHOST2"
- Section 6.2.5, "Starting the BPM Suite Components"
- Section 6.2.6, "Validating the BPM Suite Components in WLS_SOA1 and WLS_SOA2"
- Section 6.2.7, "Configuring Oracle HTTP Server for the WLS_SOAn Managed Servers"
- Section 6.2.8, "Validating Access Through Oracle HTTP Server"
- Section 6.2.9, "Backing Up the Installation"

6.2.1 Backing Up the Existing Installation

Before beginning the procedures in this section, Oracle recommends backing up the existing Fusion Middleware Home and domain. Be sure to stop the Administration Server first.

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

These commands create a backup of the installation files for both Oracle WebLogic Server and Oracle Fusion Middleware, as well as the domain configuration.

6.2.2 Running the Configuration Wizard on SOAHOST1 to Extend a SOA Domain to Include BPM

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

1. Change the directory to the location of the Configuration Wizard. This is within the SOA home directory. Domain extensions are run from the node where the Administration Server resides).

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

2. Start the Oracle Fusion Middleware Configuration Wizard:

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

3. In the Welcome screen, select **Extend an Existing WebLogic Domain**, and click **Next**.
4. In the WebLogic Domain Directory screen, select the WebLogic domain directory *ORACLE_BASE*/admin/*domain_name*/aserver/*domain_name*, and click **Next**.
5. In the Select Extension Source screen, do the following:
   - Select **Extend my domain automatically to support the following added products**. Select the following products:
   - Select the following product:
     - Oracle BPM Suite - 11.1.1.0 [soa]
6. In the Configure JDBC Component Schema screen, accept existing values (schemas created in the existing SOA system) and click **Next**.

   BPM uses the same DataSources as the existing soa-infra system.
7. In the Optional Configuration screen, select the following:
   - JMS Distributed Destinations
   - Deployments and Services

   Click **Next**.
8. In the Select JMS Distributed Destination Type screen, select UDD from the drop down list for BPMJMSModule. Leave existing modules as they are.
9. In the Target Deployments to Clusters or Servers screen, ensure the following targets:
   - Target **WSM-PM** only to **WSM-PM_Cluster**.
   - Target **usermessagingserver** and **usermessagingdriver-email** only to SOA_Cluster. (The **usermessaging-xmpp**, **usermessaging-smpp**, and **usermessaging-voicexml** applications are optional.)
   - Target the **oracle.rules.***, **oracle.sdp.***, **oracle.bpm.***, and **oracle.soa.*** deployments only to **SOA_Cluster**.
   - Target the **oracle.wsm.seedpolicies** library only to **WSM-PM_Cluster** and **SOA_Cluster**.

   Click **Next**.
10. In the Target Services to Clusters or Servers screen, ensure the following targets:

- Target **JOC Startup Class** and **JOC Shutdown Class** only to **WSM-PM_Cluster**.
- Target the **mds-owsm**, **mdw-owsm-rac0** and **mds-owsm-rac1** datasources to the **WSM-PM_Cluster** and the **AdminServer**

Click **Next**.

11. In the Configuration Summary screen click **Extend**.
12. In the Creating Domain screen, click **Done**.

    You must restart the Administration Server for this configuration to take effect. To restart the Administration Server, use the procedure in Section 4.7, "Starting the Administration Server on SOAHOST1."

6.2.3 Configuring a JMS Persistence Store for BPM JMS

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

1. Log into the Oracle WebLogic Server Administration Console.
2. In the Domain Structure window, expand the **Services** node and then click the **Persistence Stores** node.

   The Summary of Persistence Stores page appears.
3. Select the **BPMJMSFileStore_auto_1** persistence store (represented as a hyperlink) from the **Name** column of the table.

   The Settings page for the **BPMJMSFileStore_auto_1** persistence store appears.
4. In the **Configuration** tab, enter the location on a persistent storage solution (such as NAS or SAN) that is available to other servers in the cluster in the Directory field.

   Specifying this location enables pending JMS messages to be sent. The location should follow the following directory structure:

   ```
   ORACLE_BASE/admin/domain_name/soa_cluster_name/jms
   ```

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

5. Click **Save** and activate changes.
6. Repeat these steps for the **BPMJMSFileStore_auto_2** persistence store.
7. Restart the servers to make the change in the persistent stores effective.

6.2.4 Propagating the Domain Configuration to the managed server directory in SOAHOST1 and to SOAHOST2

Oracle BPM Suite requires some updates to the WebLogic Server start scripts. Propagate these changes using the pack and unpack commands.

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

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

   ```
   SOAHOST1> cd ORACLE_COMMON_HOME/common/bin

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

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

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

4. Copy the template to SOAHOST2:

   ```
   SOAHOST1> cd ORACLE_COMMON_HOME/common/bin

   SOAHOST1> scp soadomaintemplateExtBPM.jar oracle@SOAHOST2:/ORACLE_
   HOME/common/bin
   ```

5. Run the unpack command on SOAHOST2 to unpack the propagated template:

   ```
   SOAHOST2> cd ORACLE_COMMON_HOME/common/bin

   SOAHOST2> ./unpack.sh -domain=ORACLE_BASE/admin/domain_name/mserver/domain_
   name/ -overwrite_domain=true
   -template=soadomaintemplateExtBPM.jar -app_dir=ORACLE_BASE/admin/
   domain_name/mserver/applications
   ```

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

6.2.5 Starting the BPM Suite Components

For configuration changes and start scripts to be effective, you must restart the WLS_SOA*n* server to which BPM has been added.

> **Note:** Since BPM extends an already existing SOA system, the Administration Server and respective Node Managers are already running in SOAHOST1 and SOAHOST2.

To start the added BPM components:

1. Restart the WLS_SOA1 managed server:
   a. Log into the Oracle WebLogic Server Administration Console at:

      `http://ADMINVHN:7001/console`.

   b. In the Domain Structure window, expand the **Environment** node, then select **Servers**.

      The Summary of Servers page appears.

   c. Click the **Control** tab.

   d. Select **WLS_SOA1** from the **Servers** column of the table.

   e. Click **Shutdown**. Wait for the shutdown to complete (refresh the WebLogic Server Console page to verify shutdown status).

   f. Click **Start**.

2. Repeat steps a-f for **WLS_SOA2**.

6.2.6 Validating the BPM Suite Components in WLS_SOA1 and WLS_SOA2

Since the cluster address for the SOA_Cluster has already been set in the previous chapter, the BPM system can only be verified once Oracle HTTP Server has been configured to route the BPM context URLs to the WebLogic Servers. To verify the BPM system using Oracle HTTP Server, follow the steps in the sections Section 6.2.7, "Configuring Oracle HTTP Server for the WLS_SOAn Managed Servers," and Section 6.2.8, "Validating Access Through Oracle HTTP Server."

6.2.7 Configuring Oracle HTTP Server for the WLS_SOA*n* Managed Servers

To enable Oracle HTTP Server to route to BPM web applications, you must set the `WebLogicCluster` parameter to the list of nodes in the cluster:

1. Add the following lines to the *ORACLE_BASE*/ admin/*instance_name*/config/OHS/*component_name*/mod_wl_ohs.conf file:

```
# BPM
<Location /bpm/composer >
    SetHandler weblogic-handler
    WebLogicCluster SOAHOST1VHN1:8001,SOAHOST2VHN1:8001
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>

# BPM
<Location /bpm/workspace >
    SetHandler weblogic-handler
    WebLogicCluster SOAHOST1VHN1:8001,SOAHOST2VHN1:8001
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>
```

2. Restart Oracle HTTP Server on WEBHOST1 and WEBHOST2:

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

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

6.2.8 Validating Access Through Oracle HTTP Server

Verify URLs to ensure that appropriate routing and failover is working from the HTTP Server to the BPM Suite Components. To verify URLs:

1. While WLS_SOA is running, stop WLS_SOA1 using the Oracle WebLogic Server Administration Console.
2. Access `WebHost1:7777/bpm/composer` and `WebHost1:7777/bpm/workspace` to verify the appropriate functionality for BPM project Composer.
3. Start WLS_SOA1 from the Oracle WebLogic Server Administration Console.
4. Stop WLS_SOA2 from the Oracle WebLogic Server Administration Console.
5. Access `WebHost1:7777/bpm/composer` and `WebHost1:7777/bpm/workspace` to verify the appropriate functionality for BPM Workspace.

You can also verify these URLs using your load balancer address:

- `http://soa.mycompany.com:80/bpm/composer`
- `http://soa.mycompany.com:80/bpm/workspace`

6.2.9 Backing Up the Installation

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

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

1. Back up the web tier:

   a. Shut down the instance using `opmnctl`.

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

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

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

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

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

   d. Start the instance using `opmnctl`:

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

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

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

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

7 Extending the Domain to Include BAM

This chapter is for users who want to use Oracle Business Activity Monitoring (BAM) in their enterprise. It covers the following topics:

7.1 Overview of Adding BAM to a Domain

The BAM system is installed using the WL_HOME and ORACLE_HOME created in Chapter 4, "Creating a Domain" on a shared storage. BAMHOST1 and BAMHOST2 mount `MW_HOME` and reuse the existing WLS and SOA installations. The `pack` and `unpack` utilities are used to bootstrap the domain configuration for the WLS_BAM1 and WLS_BAM2 servers in these two new nodes. As a result, you do not need to install any software in these two nodes. For the BAM system to work properly, BAMHOST1 and BAMHOST2 must maintain the same system configuration that was required for installing Oracle FMW in SOAHOST1 and SOAHOST2. Otherwise, unpredictable behavior in the execution of binaries may occur.

Performing Backups

Before using the Configuration Wizard, you must back up the domain as described in in *Oracle Fusion Middleware Administrator's Guide*.

This section describes how to add BAM to the domain that you created in Chapter 4, "Creating a Domain" through the following steps.

> **Note:** You might have already added SOA components to the domain as described in Chapter 5, "Extending the Domain for SOA Components."

- Step 1: Enabling VIP4 in BAMHOST1
- Step 2: Extending the Domain to Include BAM
- Step 3: Configuring a JMS Persistence Store for BAM UMS
- Step 4: Configuring a Default Persistence Store for Transaction Recovery
- Step 5: Untargeting the BAM Server System from WLS_BAM2
- Step 6: Propagating the Domain Configuration to BAMHOST1 and BAMHOST2 Using the pack/unpack Utility
- Step 7: Starting Node Manager on BAMHOST1 and BAMHOST2
- Step 8: Starting the BAM System
- Step 9: Configuring the BAM Web Applications to Use the BAM Server in BAMHOST1
- Step 11: Configuring Oracle HTTP Server for the WLS_BAMn Managed Servers
- Step 12: Configuring Server Migration for the WLS_BAM1 Server

7.2 Enabling VIP4 in BAMHOST1

The BAM System uses a virtual hostname as the listen addresses for the managed server hosting the BAM Server component. This virtual host name and corresponding virtual IP is required to enable server migration for the BAM Server component. You must enable a VIP (VIP4) mapping BAMHOST1VHN1 on BAMHOST1 and must correctly resolve the BAMHOST1VHN1 hostname in the network system used by the topology (either by DNS Server, hosts resolution).

To enable VIP4, follow the steps described in Section 4.3, "Enabling VIP1 in SOAHOST1."

7.3 Extending the Domain to Include BAM

In this step, you extend the domain created in Chapter 4, "Creating a Domain" to contain BAM. The instructions in this section assume that the BAM deployment uses the same database service (`soaedg.mycompany.com`) as the SOA deployment. However, a deployment may choose to use a different database service specifically for BAM.

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

1. Change directory to the location of the Configuration Wizard. This is within the SOA home directory.

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

2. Start the Configuration Wizard:

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

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

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

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

   - Select **Extend my domain automatically to support the following added products**.
   - Select the following products:
     - **Oracle Business Activity Monitoring 11.1.1.0**

   Click **Next**.

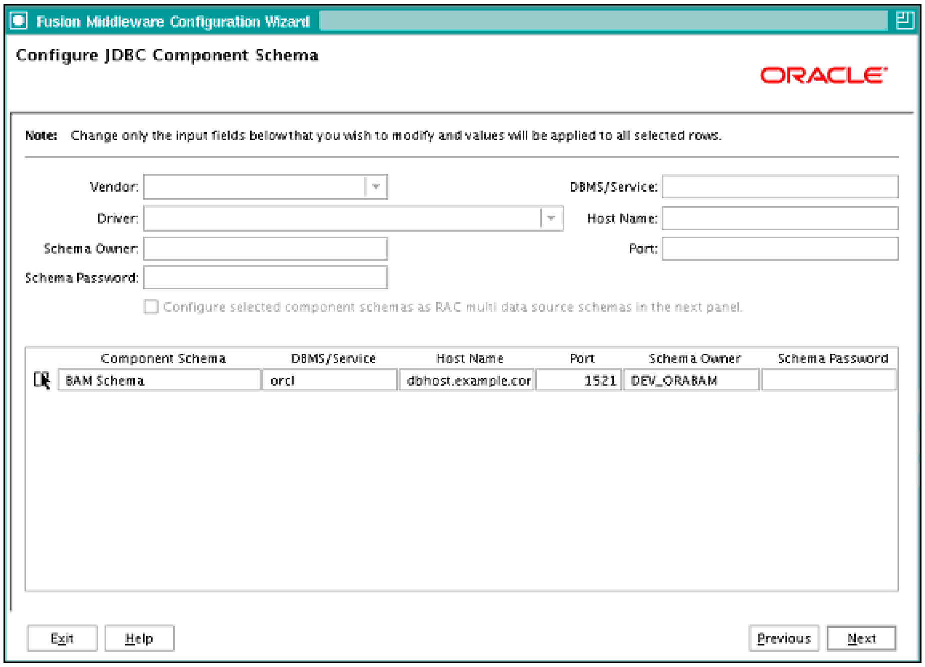
6. In the Configure JDBC Component Schema screen (Figure 7–1), do the following:

   a. Select **BAM Schema**.

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

   c. Click **Next**.

Figure 7–1 Configure JDBC Component Schema Screen



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:

Driver:

Schema Owner:

Schema Password:

DBMS/Service:

Host Name:

Port:

□ 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
BAM Schema	orcl	dbhost.example.cor	1521	DEV_ORABAM	

7. In the Configure RAC Multi Data Sources Component Schema screen (Figure 7–2), do the following:

   a. Select **BAM 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, 11**.

      – **Service Name:** Enter the service name of the database; for example, `soaedg.mycompany.com`.

      – **Username**: Enter the complete user name (including prefix) for the schemas. The user names shown in Figure 7–2 assume that `soedg` was used as prefix for schema creation from RCU.

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

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

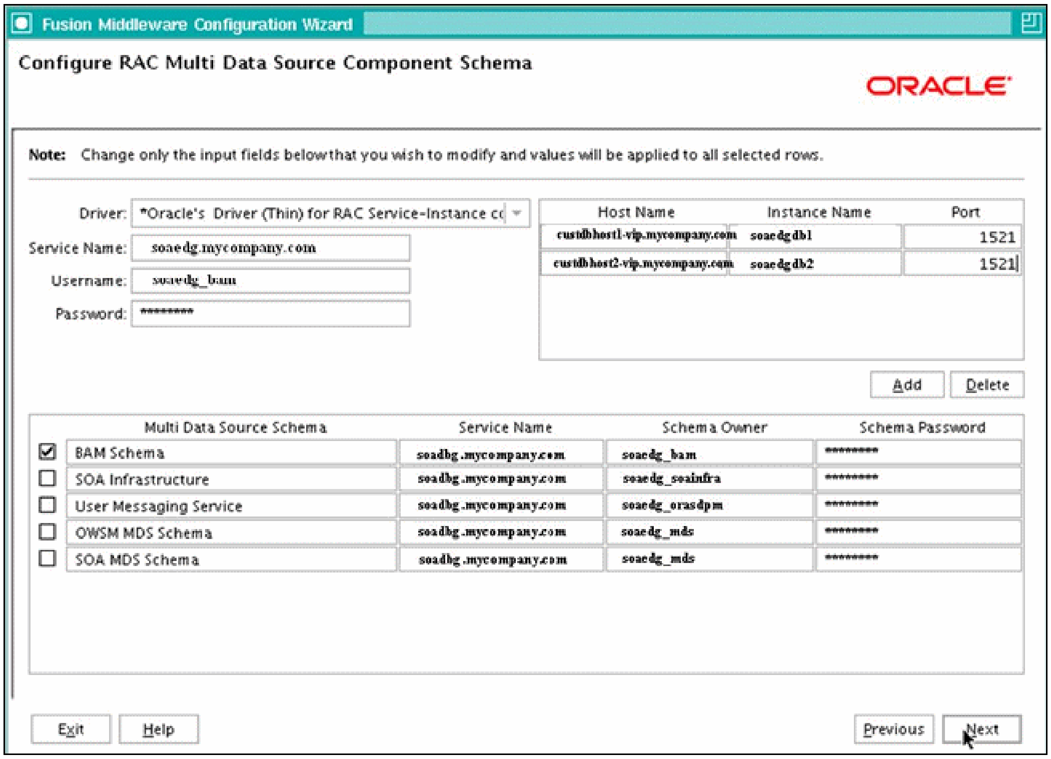
   d. Repeat for each Oracle RAC instance.

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

   ---

   e. Click **Next**.

Figure 7–2 Configure Multi Data Source Component Schema Screen



Configure RAC Multi Data Source Component Schema

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

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

Service Name: soaedg.mycompany.com

Username: soaedg_bam

Password: *********

Host Name	Instance Name	Port
custdbhost1-vip.mycompany.com	soaedgdb1	1521
custdbhost2-vip.mycompany.com	soaedgdb2	1521

	Multi Data Source Schema	Service Name	Schema Owner	Schema Password
☑	BAM Schema	soadbg.mycompany.com	soaedg_bam	*********
□	SOA Infrastructure	soadbg.mycompany.com	soaedg_soainfra	*********
□	User Messaging Service	soadbg.mycompany.com	soaedg_orasdpm	*********
□	OWSM MDS Schema	soadbg.mycompany.com	soaedg_mds	*********
□	SOA MDS Schema	soadbg.mycompany.com	soaedg_mds	*********

8. 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.

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

   - **JMS Distributed Destinations**
   - **Managed Servers, Clusters, and Machines**
   - **Deployments and Services**

10. In the Select JMS Distributed Destination Type screen, select UDD from the drop-down list for all Fusion Middleware Components' JMS Modules.

    > **Note:** Oracle does not support using WDDs for Fusion Middleware components.

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

    A server called bam_server1 is created automatically. Rename this to WLS_BAM1 and add a new server called WLS_BAM2. Give these servers the attributes listed 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_BAM1	BAMHOST1VHN1	9001	n/a	No
WLS_BAM2	BAMHOST2	9001	n/a	No

Click **Next**.

12. In the Configure Clusters screen, add the following clusters listed in Table 7–2. Do not modify the other clusters that display in this screen; leave them as they are.

Table 7–2 Clusters

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

Click **Next**.

13. In the Assign Servers to Clusters screen, add the following. Do not modify the other assignments that display in this screen; leave them as they are.

- BAM_Cluster
  - WLS_BAM1
  - WLS_BAM2

Click **Next**.

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

- Delete the **LocalMachine** that appears by default.
- Click the **Unix Machine** tab. You should add the BAMHOST1 and BAMHOST2 machines and have the following entries:

Table 7–3 Machines

Name	Node Manager Listen Address
SOAHOST1	SOAHOST1
SOAHOST2	SOAHOST2
BAMHOST1	BAMHOST1
BAMHOST2	BAMHOST2

Leave all other fields to their default values.

Click **Next**.

15. In the Assign Servers to Machines screen, do the following:

- Assign WLS_BAM1 to BAMHOST1
- Assign WLS_BAM2 to BAMHOST2.

Click **Next**.

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

- **usermessagingserver** and **usermessagingdriver-email** should be targeted only to **SOA_Cluster** and **BAM_Cluster**. (The usermessaging-xmpp, usermessaging-smpp, and usermessaging-voicexml applications are optional.)
- **WSM-PM** should be targeted only to **WSM-PM_Cluster**.
- The **DMS Application** should be targeted to **BAM_Cluster**, **SOA_Cluster**, **WSM-PM_Cluster** and **Admin Server**.
- The **oracle.sdp.*** deployment should be targeted only to **SOA_Cluster** and **BAM_Cluster**. The **oracle.soa.*** deployments should be targeted only to **SOA_Cluster**.
- The **oracle.rules.*** library should be targeted to the **SOA_Cluster**, **BAM_Cluster**, and **Admin Server**.
- The **oracle.wsm.seedpolicies** library should be targeted only to the **WSM-PM_Cluster**.

  Target this library to the **SOA_Cluster** or **BAM_Cluster** also, only if you are planning to deploy WebLogic WebServices to them.
- **oracle.bam*** is targeted only to **BAM_Cluster**.

Click **Next**.

17. In the Target Services to Clusters or Servers screen, ensure the following targets:
    - **mds-owsm**, **mds-owsm-rac0**, and **mds-owsm-rac1** should be targeted to both **WSM-PM_Cluster** and **AdminServer**.
    - **mds-soa**, **mds-soa-rac0**, and **mds-soa-rac1** should be targeted to both **SOA_Cluster** and **AdminServer**.
    - **OraSDPMDatasource**, **OraSDPMDatasource-rac0**, and **OraSDPMDatasource-rac1** should be targeted to both **SOA_Cluster** and **BAM_Cluster**.
    - Target **JOC Startup Class** and **JOC Shutdown Class** only to **WSM-PM_Cluster.**
18. Click **Next**.
19. In the Configuration Summary screen, click **Extend**.
20. Click **OK** in the warning dialog about conflicts in ports for the domain.
21. In the Creating Domain screen, click **Done**.
22. Restart the Administration Server to enable these changes to take effect.

7.3.1 Restarting the Administration Server

Restart the Administration Server using the procedure in Section 4.7, "Starting the Administration Server on SOAHOST1."

7.4 Configuring a JMS Persistence Store for BAM UMS

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

1. Log into the Oracle WebLogic Server Administration Console.

2. In the Domain Structure window, expand the **Services node** and then click the **Persistence Stores** node. The Summary of Persistence Stores page appears.
3. Select the **UMSJMSFileStore_auto_3** persistence store (represented as a hyperlink) from the Name column of the table. The Settings page for the persistence store appear.
4. In the Configuration tab, enter the location on a persistent storage solution (such as NAS or SAN) that is available to other servers in the cluster in the Directory field. Specifying this location enables pending JMS messages to be sent. The location should follow the following directory structure:

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

5. Click **Save and Activate**.
6. Repeat the steps for **UMSJMSFileStore_auto_4**

7.5 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 the server.

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

To set the location for the default persistence store, complete these steps for WLS_BAM1:

1. Log into the Oracle WebLogic Server Administration Console.
2. In the Domain Structure window, expand the **Environment** node and then click the **Servers** node. The Summary of Servers page appears.
3. Click **WLS_BAM1** (represented as a hyperlink) in the Name column of the table. The settings page for the WLS_BAM1 server appears and defaults to the **Configuration** tab.
4. Click the **Services** tab.
5. In the Default Store section of the page, enter the path to the folder where the default persistent stores will store its data files. The directory structure of the path is as follows:

   ```
   ORACLE_BASE/admin/domain_name/bam_cluster_name/tlogs
   ```

6. Click **Save**.

> **Note:** To enable migration of the Transaction Recovery Service, specify a location on a persistent storage solution that is available to other servers in the cluster. Both BAMHOST1 and BAMHOST2 must be able to access this directory. This directory must also exist before you restart the server.

7.6 Untargeting the BAM Server System from WLS_BAM2

Because the BAM server component in BAM is a singleton, you must untarget it from one of the WLS_BAM servers before you configure it for server migration.

In this step, you remove the BAM server runtime from WLS_BAM2. Perform these steps to untarget the BAM server artifacts from WLS_BAM2:

1. Log into the Oracle WebLogic Administration Console at `http://ADMINVHN:7001/console`.
2. In the Domain Structure window, choose **Environment** and then **Servers**. The Summary of Servers page appears.
3. Select **WLS_BAM2** in Name column of the table. The Settings page for WLS_ BAM2 appears.
4. Click the **Deployments** tab.
5. Select the **oracle-bam** application from the Name column of the table. The Settings page for the oracle-bam application appears.
6. Click the **Targets** tab.
7. Click **Lock and Edit**.
8. Change the targets for the modules as described in Table 7–4.

> **Note:** You must target all of these components as described in Table 7–4, as incorrect targeting can prevent the BAM system from starting.

Table 7–4 oracle-bam Component Target Types

Component	Type	Target
oracle-bam(11.1.1)	Enterprise Application	BAM_Cluster
/oracle/bam	WEBAPP	WLS_BAM1
oracle-bam-adc-ejb.jar	EJB	WLS_BAM1
oracle-bam-ems-ejb.jar	EJB	WLS_BAM1
oracle-bam-eventengine-ejb.jar	EJB	WLS_BAM1
oracle-bam-reportcache-ejb.jar	EJB	WLS_BAM1
OracleBAM	WEBAPP	BAM_Cluster
OracleBAMWS	WEBAPP	BAM_Cluster
oracle-bam-statuslistener-ejb.jar	EJB	WLS_BAM1
sdpmessagingclient-ejb.jar	EJB	WLS_BAM1

9. Click **Save** and **Activate Changes**.

7.7 Propagating the Domain Configuration to BAMHOST1 and BAMHOST2 Using the pack/unpack Utility

To propagate the new domain configuration, complete these steps:

1. Make sure that a similar directory and shared storage configuration as SOAHOST2 is present in BAMHOST1 (described in Chapter 2, "Database and Environment Preconfiguration"). BAMHOST1 and BAMHOST2 should have mounted the `MW_HOME` directory as created in Chapter 4, "Creating a Domain."

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

   a. Run the following command:

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

   > **Note:** Notice that this directory is available as mount point to the MW_HOME created in Chapter 4, "Creating a Domain."

   b. Run the pack command:

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

3. Run the following command on SOAHOST1 to copy the template file created in the previous step to BAMHOST1.

   ```
   SOAHOST1> scp soadomaintemplateBAM.jar
      oracle@BAMHOST1:/ORACLE_COMMON_HOME/common/bin
   ```

4. Run the `unpack` command on BAMHOST1 to unpack the template in the managed server domain directory as follows:

   ```
   BAMHOST1> cd ORACLE_COMMON_HOME/common/bin
   BAMHOST1> ./unpack.sh -domain= ORACLE_BASE/admin/
   domain_name/mserver/domain_name -template=soadomaintemplateBAM.jar
   -app_dir=ORACLE_BASE/admin/domain_name/mserver/applications
   ```

5. Run the copy and unpack commands for BAMHOST2.

7.8 Disabling Host Name Verification for the WLS_BAM*n* 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 8, "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 Enterprise Deployment topology configuration is complete as described in Chapter 8, "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_BAM1**, then **SSL**, and then **Advanced**.
3. Set Hostname Verification to **None**.
4. In the Administration Console, select **WLS_BAM2**, then **SSL**, and then **Advanced**.
5. Save and activate the changes.

7.9 Starting Node Manager on BAMHOST1 and BAMHOST2

Perform these steps to start Node Manager on BAMHOST1 and BAMHOST2:

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:

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

   > **Note:** You must use the `StartScriptEnabled` property to avoid class loading failures and other problems. See also Section 10.7.5, "Incomplete Policy Migration After Failed Restart of SOA Server."

   > **Note:** If the BAM 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 startscript.

2. Run the following commands to start Node Manager on BAMHOST1:

   ```
   BAMHOST1> cd WL_HOME/server/bin
   BAMHOST1> ./startNodeManager.sh
   ```

   Run the following commands to start Node Manager on BAMHOST2:

   ```
   BAMHOST2> cd WL_HOME/server/bin
   BAMHOST2> ./startNodeManager.sh
   ```

7.10 Starting the BAM System

Perform these steps to start the WLS_BAM1 managed server on BAMHOST1:

1. Start the WLS_BAM1 managed servers:

   a. Log into 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 appears.

   c. Click the **Control** tab.

   d. Select **WLS_BAM1** from the Servers column of the table.

   e. Click **Start**.

2. Access `http://BAMHOST1VHN1:9001/OracleBAM` to verify status of WLS_BAM1.

If the managed server fails to start with the following message:

```
Listener refused the connection with the following error:
ORA-12519, TNS:no appropriate service handler found
The Connection descriptor used by the client was <db_connect_string>
```

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 the first server.

1. Start the WLS_BAM2 managed servers:
   a. Log into the Oracle WebLogic 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 appears.
   c. Click the **Control** tab.
   d. Select **WLS_BAM2** from the Servers column of the table.
   e. Click **Start**.
2. Access `http://BAMHOST2:9001/OracleBAM` to verify status of WLS_BAM2.

> **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.

7.11 Configuring the BAM Web Applications to Use the BAM Server in BAMHOST1

Perform these steps to configure the OracleBamWeb(WLS_BAM1) and OracleBamWeb(WLS_BAM2) applications to use the BAM server in BAMHOST1:

1. Access Oracle Enterprise Manager Fusion Middleware Control through `http://ADMINVHN:7001/em`.
2. Expand **BAM** in the navigation tree.
3. Right-click **OracleBamWeb(WLS_BAM1)**.
4. Choose **BAM Web Properties** from the context menu. The BAM Web Properties page appears.
5. Define the following properties:
   - Enter **https://soa.mycompany.com:443** for the application URL.
   - Enter *BAMHOST1VHN1* for the server name. See also Table 2–2 in Section 2.2.3, "IPs and Virtual IPs."
   - Modify the listening port of the server using the Mbean browser. To do this follow these steps:
     – Log into the Oracle Enterprise Manager Fusion Middleware Control.
     – Expand the domain name in the left navigation tree.
     – Expand the BAM item in the left navigation tree.
     – In the BAM drop-down menu on the top-right, select **Mbean Browser**.
     – Navigate to the **oracle.bam.web**, **Server**, **Application**, **Config**, and then **BAMWebConfig**, on the right.
     – In the **ServerPort** field, replace the "DEFAULT" value with **9001**.

6. Select **OracleBamWeb(WLS_BAM2)** from the navigation tree and repeat these steps.

7.12 Configuring Oracle HTTP Server for the WLS_BAM*n* Managed Servers

To enable Oracle HTTP Server to route to BAM_Cluster, which contains the WLS_BAM*n* managed servers, you must set the `WebLogicCluster` parameter to the list of nodes in the cluster as follows:

1. Add the following lines to the `ORACLE_BASE/admin/ohs_instance1/config/OHS/ohs1/mod_wl_ohs.conf` file:

```
# BAM Web Application
<Location /OracleBAM >
    SetHandler weblogic-handler
    WebLogicCluster BAMHOST1VHN1:9001,BAMHOST2:9001
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>

<Location /OracleBAMWS >
    SetHandler weblogic-handler
    WebLogicCluster BAMHOST1VHN1:9001,BAMHOST2:9001
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>
```

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

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

7.13 Validating Access Through Oracle HTTP Server

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

1. While WLS_BAM2 is running, stop WLS_BAM1 using the Oracle WebLogic Server Administration Console.
2. Access `WebHost1:7777/OracleBAM` to verify the appropriate functionality. (Please note that you will not be able to retrieve reports or data since the BAM server is down.)
3. Start WLS_BAM1 from the Oracle WebLogic Server Administration Console.
4. Stop WLS_BAM2 from the Oracle WebLogic Server Administration Console.
5. Access `WebHost1:7777/OracleBAM` to verify the appropriate functionality.

7.14 Configuring Server Migration for the WLS_BAM1 Server

The high-availability architecture for BAM uses server migration to protect the BAM server singleton service against failures. The WLS_BAM1 managed server is configured so that it can be restarted on BAMHOST2 if it fails. For this configuration, WLS_BAM1 listens on a specific, floating IP address that is failed over by WebLogic

Server migration. To configure server migration for the WLS_BAM1 managed servers, complete the following tasks:

- Step 1: Setting Up the User and Tablespace for the Server Migration Leasing Table
- Step 2: Creating a Multi-Data Source from the WebLogic Server Administration Console
- Step 3: Editing the Node Manager's Properties File
- Step 4: Set Environment and Superuser Privileges for the wlsifconfig.sh Script
- Step 5: Enabling Host Name Verification Certificates Between Node Manager in the BAMHOSTn Nodes and the Administration Server
- Step 6: Configure Server Migration Targets
- Step 7: Test Server Migration

> **Note:** If server migration was configured previously for SOA, the BAM stem can use the same data sources and database schemas. In that case, steps 1 through 4 may not be required, but you must also target the corresponding server-migration/leasing datasources to the BAM Cluster.

7.14.1 Setting Up the User and Tablespace for the Server Migration Leasing Table

Perform these steps to create the user and tablespace:

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 it to the leasing tablespace as follows:

```
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.dll` script as follows:

   a. Copy the *leasing.ddl* file located in 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 as follows:

   ```
   SQL> @copy_location/leasing.ddl;
   ```

7.14.2 Creating a Multi-Data Source from the WebLogic Server Administration Console

Use the Oracle WebLogic Server Administration Console to create a multi-data source for the leasing table. 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 for the global leasing multi-data source. When you create this data source:

- Ensure that it 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/participation algorithm for the data source (do not choose the **Supports Global Transactions** option, or 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 BAM Cluster.
- Make sure the datasources' connection pool initial capacity is set to **0**. To do this, select **Services**, **JDBC**, and then **Datasources**. In the Datasources screen, click the **Datasource Name**, then click the **Connection Pool** tab, and enter **0** in the **Initial capacity** field.

For information on using Oracle WebLogic Server Administration to create a multi-data source, see Section 9.2, "Creating a Multi-Data Source Using the Oracle WebLogic Server Administration Console."

7.14.3 Editing the Node Manager's Properties File

The `nodemanager.properties` file is located in the *WL_HOME*/common/nodemanager directory.

Add the following properties to enable server migration to work properly:

```
Interface=eth0
NetMask=255.255.255.0
UseMACBroadcast=true
```

- `Interface`

  This property specifies the interface name for the floating IP (`eth0`, for example).

  > **Note:** Do not specify the sub interface, such as `eth0:1` or `eth0:2`. This interface is to be used without the `:0`, or `:1`. The 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`, or, `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.

- `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.

Perform this configuration in the two nodes where the servers are running. Verify in the output of Node Manager (the shell where the Node Manager is started) that these properties are in use. Otherwise, problems may occur during migration. The output should be similar to the following:

```
...
StateCheckInterval=500
```

```
Interface=eth0
NetMask=255.255.255.0
...
```

> **Note:** The following steps are not required if the server properties (start properties) have been properly set and Node Manager can start the servers remotely.

1. If not done already, set the `StartScriptEnabled` property in the `nodemanager.properties` file to true. This is required to enable Node Manager to start the managed servers.
2. Start Node Manager on Node 1 and Node 2 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 SOAHOSTn, use the `Interface` environment variable as follows: `SOAHOSTn> export JAVA_OPTIONS=-DInterface=eth3` and start Node Manager after the variable has been set in the shell.

7.14.4 Set Environment and Superuser Privileges for the wlsifconfig.sh Script

Perform these steps to set the environment and superuser privileges for the `wlsifconfig.sh` script:

1. Ensure that the PATH environment variable includes the files listed in Table 7–5.

Table 7–5 Required Files for the PATH Environment

File	Directory Location
`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, to set the environment and superuser privileges for the `wlsifconfig.sh` script, complete these steps:
     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.

7.14.5 Enabling Host Name Verification Certificates Between Node Manager in the BAMHOST*n* Nodes and the Administration Server

See Section 8.3, "Enabling Host Name Verification Certificates for Node Manager in SOAHOST1."

7.14.6 Configure Server Migration Targets

Configuring Cluster Migration sets the `DataSourceForAutomaticMigration` property to true. Perform these steps to configure cluster migration in a cluster:

1. Log into the Oracle WebLogic Server Administration Console.
2. In the Domain Structure window, expand **Environment** and select **Clusters**. The Summary of Clusters page appears.
3. Click the cluster for which you want to configure migration (**BAM_Cluster**) in the Name column of the table.
4. Click the **Migration** tab.
5. In the **Available** field, select the machine to which to allow migration and click the right arrow. In this case, select **BAMHOST1** and **BAMHOST2**.
6. Select the data source to be used for automatic migration. In this case select the leasing data source.
7. Click **Save**.
8. Set the candidate machines for server migration:

Note: You must perform this task for WLS_BAM1 only, as WLS_BAM2 does not use server migration.

---

a. In Domain Structure window of the Oracle WebLogic Server Administration Console, expand **Environment** and select **Servers**.

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. Select **BAMHOST2** for **WLS_BAM1**.

e. Select **Automatic Server Migration Enabled**. This enables the Node Manager to start a failed server on the target node automatically.

f. Click **Save**.

g. Restart the Administration Server and the WLS_BAM1 server.

To restart the Administration Server, use the procedure in Section 4.7, "Starting the Administration Server on SOAHOST1.".

7.14.7 Test Server Migration

Perform these steps to verify that Server Migration is working properly:

From Node 1:

1. Kill the WLS_BAM1 managed server.

   To do this, run this command:

   ```
   BAMHOST1> 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:

   ```
   BAMHOST1> ps -ef | grep WLS_BAM1
   ```

2. Watch the Node Manager console: you should see a message indicating that WLS_BAM1's floating IP has been disabled.
3. Wait for the Node Manager to try a second restart of WLS_BAM1. Node Manager waits for a fence period of 30 seconds before trying this restart.
4. Once Node Manager restarts the server, stop it again. Now Node Manager should log a message indicating that the server will not be restarted again locally.

From Node 2:

1. Watch the local Node Manager console. After 30 seconds since the last try to restart WLS_BAM1on Node 1, Node Manager on Node 2 should prompt that the floating IP for WLS_BAM1 is being brought up and that the server is being restarted in this node.
2. Access the Oracle BAM console using BAMHOST1VHN1 and soa.mycompany.com/OracleBAM.

Verification From the Administration Console

Migration can also be verified in the Administration Console:

1. Log into the Administration Console.
2. Click on **Domain** on the left console.
3. Click the **Monitoring** tab and then on the **Migration** tab.

   The Migration Status table provides information on the status of the migration.

7.15 Configuration Changes Applied to BAM components in an Enterprise Deployment Topology

If you are using Oracle BAM in a clustered environment, any configuration property changes you make in Oracle Enterprise Manager on one node must be made on all nodes (i.e. configuration changes applied to one server are no applied automatically to all servers in the BAM cluster. In addition, consider the following when making configuration changes to BAM Server in a BAM Enterprise Deployment Topology:

Since server migration is used, the BAM Server is moved to a different node's domain directory. It is necessary to pre-create the BAM Server configuration in the failover node. The BAM Server configuration files are located in the following directory:

DOMAIN_HOME/config/fmwconfig/servers/*server_name*/applications/oracle_bam-11.1.1/config

In order to create the files in preparation for possible failovers, you can force a server migration and copy the files from the source node. For example, for BAM:

1. Configure the driver for WLS_BAM1 in BAMHOST1.
2. Force a failover of WLS_BAM1 to BAMHOST2. Verify the directory structure for the BAM Server in the failover node:

   ```
   DOMAIN_HOME/config/fmwconfig/servers/server_name/applications/oracle_
   bam-11.1.1/config
   ```

7.16 Accessing an Oracle BAM Server Using the BAM Adapter

When accessing an Oracle BAM Server using the BAM Adapter (rmi), the Virtual Hostname of the BAM Server (BAMHOST1VNH1) should be used for the connection. SOAP requests come through HTTP, therefore, you must use the load balancer addresses when using the adapter in this case.

7.17 Backing Up the Installation

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

Perform these steps to back up the installation a 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 database. This is a full database backup (either hot or cold) using Oracle Recovery Manager (recommended) or OS tools such as `tar` for cold backups if possible.

3. Back up the AdminServer domain directory. Perform a backup to save your domain configuration. The configuration files all exist under the `ORACLE_BASE/admin/<domain_name>` directory.

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

8 Setting Up Node Manager

This chapter describes how to configure Node Manager per the Enterprise Deployment recommendations. Oracle recommends using host name verification for the communications between Node Manager and the Administration Server. This requires the use of certificates for the different addresses communicating with the Administration Server. In this chapter, the steps for configuring SOAHOST1 and SOAHOST2 certificates for host name verification are provided. Similar steps would be required for BAMHOST1 and BAMHOST2 in a BAM Enterprise Deployment topology. Although the appropriate hos tname changes in the steps are required for BAM, the procedure and syntax are exactly the same.

Oracle also recommends placing your Oracle Fusion Middleware deployment’s NodeManager's log in a different location from the default (which is inside the MW_ Home where Node Manager is located). See Section 8.2, "Changing the Location of Node Manager Log" for details.

This chapter includes the following sections:

- Section 8.1, "About the Node Manager"
- Section 8.2, "Changing the Location of Node Manager Log"
- Section 8.3, "Enabling Host Name Verification Certificates for Node Manager in SOAHOST1"
- Section 8.4, "Starting the Node Manager on SOAHOST1"
- Section 8.5, "Enabling Host Name Verification Certificates for the Node Manager in SOAHOST2"
- Section 8.6, "Starting the Node Manager on SOAHOST2"
- Section 8.7, "Configuring WebLogic Servers to Use the Custom Keystores"

8.1 About the Node Manager

The Node Manager enables you to start and stop the Administration Server and the managed servers.

About Passwords

The passwords used in this guide are used only as examples. Use secure passwords in a production environment. For example, use passwords that include both uppercase and lowercase characters as well as numbers.

8.2 Changing the Location of Node Manager Log

Edit the `nodemanager.properties` file located in the *MW_HOME*/wlserver_10.3/common/nodemanager directory. Add the new location for the LogFile. Oracle recommends locating this file out of the MW_HOME directory, and inside the admin directory for the deployment using the following command:

```
LogFile=ORACLE_BASE/admin/nodemanager.log
```

Restart Node Manager for the change to take effect.

8.3 Enabling Host Name Verification Certificates for Node Manager in SOAHOST1

Perform these steps to set up host name verification certificates for communication between the Node Manager and the Administration Server.

- Step 1: Generating Self-Signed Certificates Using the utils.CertGen Utility
- Step 2: Creating an Identity Keystore Using the utils.ImportPrivateKey Utility
- Step 3: Creating a Trust Keystore Using the Keytool Utility
- Step 4: Configuring Node Manager to Use the Custom Keystores

8.3.1 Generating Self-Signed Certificates Using the utils.CertGen Utility

Follow these steps to create self-signed certificates on SOAHOST1.mycompany.com. These certificates should be created using the network name/alias. When a server is using a virtual hostname 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 be located on a shared storage that is accessible from the failover node. For information on using trust CA certificates instead, see "Configuring Identity and Trust" in *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

1. Set up your environment by running the `WL_HOME/server/bin/setWLSEnv.sh` script:

   In the Bourne shell, run the following command:

   ```
   SOAHOST1> . setWLSEnv.sh
   ```

   Verify that the CLASSPATH environment variable is set:

   ```
   SOAHOST1> echo $CLASSPATH
   ```

2. 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 (for example, SSL set up for HTTP invocations). In this case, SOAHOST2 uses the cert directory created for SOAHOST1 certificates. Create a user-defined directory for the certificates:

   ```
   SOAHOST1> mkdir certs
   ```

3. Change directory to the user-defined directory.

   ```
   SOAHOST1> cd certs
   ```

4. Run the utils.CertGen tool from the user-defined directory to create the certificates for both *HOST*. mycompany.com and *VIP*. mycompany.com.

   Syntax:

   `java utils.CertGen` *`key_passphrase cert_file_name key_file_name`* `[export | domestic] [hostname]`

   Examples:

   ```
   SOAHOST1> java utils.CertGen welcome1 SOAHOST1.mycompany.com_cert
   SOAHOST1.mycompany.com_key domestic SOAHOST1.mycompany.com

   SOAHOST1> java utils.CertGen welcome1 ADMINVHN.mycompany.com_cert
   ADMINVHN.mycompany.com_key domestic ADMINVHN.mycompany.com
   ```

8.3.2 Creating an Identity Keystore Using the utils.ImportPrivateKey Utility

Follow these steps to create an Identity Keystore on SOAHOST1.mycompany.com.

The procedures described in the previous sections created an Identity keystore that resides in a shared storage. In this section new keys for SOAHOST2 are added to the store. Import the certificate and private key for both SOAHOST2 and SOAHOST2VHN1 into the Identity Store. Make sure you use a different alias for each of the certificate/key pairs imported.

1. Create a new identity keystore called *appIdentityKeyStore* using the utils.ImportPrivateKey utility.

   Create this keystore under the same directory as the certificates (that is, `ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>/certs`.

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

   Import the certificate and private key for both SOAHOST1 and VIPHOST1 into the Identity Store. Make sure that you use a different alias for each of the certificate/key pair imported.

   Syntax:

   `java utils.ImportPrivateKey` *`<keystore_file> <keystore_password> <certificate_alias_to_use> <private_key_passphrase> <certificate_file> <private_key_file>`* `[`*`<keystore_type>`*`]`

   Examples:

   ```
   SOAHOST1> java utils.ImportPrivateKey appIdentityKeyStore.jks welcome1
             appIdentity1 welcome1
             ORACLE_BASE/admin/domain_name/aserver/domain_name/certs/SOAHOST1_
   cert.pem
             ORACLE_BASE/admin/domain_name/aserver/domain_name/certs/SOAHOST1_
   key.pem

   SOAHOST1> java utils.ImportPrivateKey appIdentityKeyStore.jks welcome1
             appIdentity2 welcome1
             ORACLE_BASE/admin/domain_name/aserver/domain_
   name/certs/SOAHOST1VHN1_cert.pem
             ORACLE_BASE/admin/domain_name/aserver/domain_
   ```

```
name/certs/SOAHOST1VHN1_key.pem

SOAHOST1> java utils.ImportPrivateKey appIdentityKeyStore.jks welcome1
             appIdentity3 welcome1
             ORACLE_BASE/admin/domain_name/aserver/domain_name/certs/ADMINVHN_
cert.pem
             ORACLE_BASE/admin/domain_name/aserver/domain_name/certs/ADMINVHN_
key.pem
```

8.3.3 Creating a Trust Keystore Using the Keytool Utility

Follow these steps to create the Trust Keystore on SOAHOST1.mycompany.com.

1. Copy the standard java keystore to create the new trust keystore since it already contains most of the root CA certificates needed. Oracle does not recommend modifying the standard Java trust key store directly. Copy the standard Java keystore CA certificates located under the *WL_HOME*/server/lib directory to the same directory as the certificates. For example:

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

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

   ```
   keytool -storepasswd -new <NewPassword> -keystore <TrustKeyStore> -storepass
   <Original Password>
   ```

   For example:

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

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

   ```
   keytool -import -v -noprompt -trustcacerts -alias <AliasName>
    -file <CAFileLocation> -keystore <KeyStoreLocation> -storepass <KeyStore
   Password>
   ```

   For example:

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

8.3.4 Configuring Node Manager to Use the Custom Keystores

To configure the Node Manager to use the custom keystores, add the following lines to the end of the `nodemanager.properties` file located in the `WL_HOME/common/nodemanager` directory:

```
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityKeyStoreFileName=<Identity KeyStore>
CustomIdentityKeyStorePassPhrase=<Identity KeyStore Passwd>
CustomIdentityAlias=<Identity Key Store Alias>
CustomIdentityPrivateKeyPassPhrase=<Private Key used when creating Certificate>
```

```
Make sure to use the correct value for CustomIdentityAlias for Node Manager
listen's address. For example on SOAHOST1, use appIdentity1 as per the above steps
 (appIdentity1 mapped to the SOAHOST1 listen address).
Example for Node 1:
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityKeyStoreFileName=ORACLE_BASE/admin/<domain_name>/aserver/<domain_
name>/certs/appIdentityKeyStore.jks
CustomIdentityKeyStorePassPhrase=welcome1
CustomIdentityAlias=appIdentity1
CustomIdentityPrivateKeyPassPhrase=welcome1
```

The passphrase entries in the `nodemanager.properties` file get encrypted when you start Node Manager as described in Section 8.4, "Starting the Node Manager on SOAHOST1." For security reasons, you want to minimize the time the entries in the `nodemanager.properties` file are left unencrypted. After you edit the file, you should start Node Manager as soon as possible so that the entries get encrypted.

When using a common/shared storage installation for MW_HOME, Node Manager is started from different nodes using the same base configuration (nodemanager.properties). In that case, it is required to add the certificate for all the nodes that share the binaries to the appIdentityKeyStore.jks identity store. To do this, create the certificate for the new node and import it to appIdentityKeyStore.jks as described above. Once the certificates are available in the store, each node manager needs to point to a different identity alias to send the correct certificate to the Administration Server. To do this, set different environment variables before starting Node Manager in the different nodes:

```
SOAHOST1> cd WL_HOME/server/bin
SOAHOST1>export JAVA_OPTIONS=-DCustomIdentityAlias=appIdentitySOAHOST1

SOAHOSTn> cd WL_HOME/server/bin
SOAHOSTn> export JAVA_OPTIONS=-DCustomIdentityAlias=appIdentitySOAHOSTn
```

8.4 Starting the Node Manager on SOAHOST1

Run these commands to start Node Manager on SOAHOST1:

> **Note:** If you have not configured and started Node Manager for the first time yet, run the *setNMProps.sh* script as specified in section Section 4.6, "Starting Node Manager on SOAHOST1." This will enable the use of the start script which is required for SOA and/or BAM.

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

8.5 Enabling Host Name Verification Certificates for the Node Manager in SOAHOST2

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

- Step 1: Generating Self-Signed Certificates Using the utils.CertGen Utility
- Step 2: Importing Identities in SOAHOST2 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

8.5.1 Generating Self-Signed Certificates Using the utils.CertGen Utility

Follow these steps to create self-signed certificates on SOAHOST2.mycompany.com. These certificates should be created using the network name/alias.

1. Set up your environment by running the `WL_HOME/server/bin/setWLSEnv.sh` script:

   In the Bourne shell, run the following command:

   ```
   SOAHOST2> . setWLSEnv.sh
   ```

   Verify that the CLASSPATH environment variable is set:

   ```
   SOAHOST2> echo $CLASSPATH
   ```

2. The directory where keystores and trust keystores are maintained must be on shared storage that is accessible from all nodes so that when the Administration Server or SOA servers fail over, (manually or with server migration), the appropriate certificates can be accessed. In this case, SOAHOST2 uses the cert directory created for SOAHOST1 certificates. If you are maintaining duplicated stores, create user-defined directory for the certificates.

   ```
   SOAHOST2> mkdir certs
   ```

3. Change directory to the user-defined directory.

   ```
   SOAHOST2> cd certs
   ```

4. Run the utils.CertGen tool from the user-defined directory to create the certificates for both SOAHOST2 and SOAHOST2VHN1.

   Syntax:

   ```
   java utils.CertGen key_passphrase cert_file_name key_file_
   name [export | domestic] [hostname]
   ```

   Examples:

   ```
   SOAHOST2> java utils.CertGen welcome1 SOAHOST2_cert SOAHOST2_key
                domestic SOAHOST2.mycompany.com

   SOAHOST2> java utils.CertGen welcome1 SOAHOST2VHN1_cert SOAHOST2VHN1_key
                domestic SOAHOST2VHN1.mycompany.com
   ```

8.5.2 Importing Identities in SOAHOST2 using the "utils.ImportPrivateKey" Utility

Follow these steps to create an Identity Keystore on SOAHOST2.mycompany.com.

The procedures described in the previous sections created an Identity keystore that resides in a shared storage. In this section new keys for SOAHOST2 are added to the store. Import the certificate and private key for both SOAHOST2 and SOAHOST2VHN1 into the Identity Store. Make sure you use a different alias for each of the certificate/key pairs imported.

Syntax:

```
java utils.ImportPrivateKey <keystore_file> <keystore_password>
<certificate_alias_to_use> <private_key_passphrase>
<certificate_file> <private_key_file> [<keystore_type>]
```

Examples:

```
SOAHOST2> java utils.ImportPrivateKey appIdentityKeyStore.jks welcome1
            appIdentity3 welcome1
            ORACLE_BASE/admin/domain_name/aserver/domain_name/certs/SOAHOST2_
cert.pem
            ORACLE_BASE/admin/domain_name/aserver/domain_name/certs/SOAHOST2_
key.pem

SOAHOST2> java utils.ImportPrivateKey appIdentityKeyStore.jks welcome1
            appIdentity4 welcome1
            ORACLE_BASE/admin/domain_name/aserver/domain_name/certs/SOAHOST2VHN1_
cert.pem
            ORACLE_BASE/admin/domain_name/aserver/domain_name/certs/SOAHOST2VHN1_
key.pem
```

8.5.3 Configuring Node Manager to Use the Custom Keystores

Follow these steps to configure the Node Manager to use the custom keystores.

1. Add the following lines to the end of the `nodemanager.properties` file located in the `WL_HOME/common/nodemanager` directory.

   ```
   KeyStores=CustomIdentityAndCustomTrust

   CustomIdentityKeyStoreFileName=<Identity KeyStore>

   CustomIdentityKeyStorePassPhrase=<Identity KeyStore Passwd>

   CustomIdentityAlias=<Identity Key Store Alias>

   CustomIdentityPrivateKeyPassPhrase=<Private Key used when
   creating Certificate>
   ```

   Make sure to use the correct value for CustomIdentityAlias on each node. For example, on SOAHOST2, use "appIdentity3".

   Example for Node 2:

   ```
   KeyStores=CustomIdentityAndCustomTrust

   CustomIdentityKeyStoreFileName=ORACLE_BASE/admin/<domain_
   name>/aserver/<domain_name>/certs/appIdentityKeyStore.jks

   CustomIdentityKeyStorePassPhrase=welcome1

   CustomIdentityAlias=appIdentity3

   CustomIdentityPrivateKeyPassPhrase=welcome1
   ```

   > **Note:** The passphrase entries in the `nodemanager.properties` file get encrypted when you start Node Manager, as described in Section 8.6, "Starting the Node Manager on SOAHOST2."
   >
   > For security reasons, you want to minimize the time the entries in the `nodemanager.properties` file are left unencrypted. After you edit the file, you should start Node Manager as soon as possible so that the entries get encrypted.

8.6 Starting the Node Manager on SOAHOST2

Run these commands to start Node Manager on SOAHOST2:

Note: If you have not configured and started Node Manager for the first time yet, run the *setNMProps.sh* script as specified in section Section 4.15, "Starting Node Manager on SOAHOST2." This will enable the use of the start script which is required for SOA and/or BAM.

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

8.7 Configuring WebLogic Servers to Use the Custom Keystores

To configure the identity and trust keystores:

1. Log in to the Administration Console, and click **Lock & Edit**.
2. In the left pane, expand **Environment**, and select **Servers**.
3. Click the name of the server for which you want to configure the identity and trust keystores.
4. Select **Configuration**, and then **Keystores**.
5. In the **Keystores** field, select the "**Custom Identity and Custom Trust**" method for storing and managing private keys/digital certificate pairs and trusted CA certificates.
6. In the **Identity** section, define attributes for the identity keystore.

   a. **Custom Identity Keystore**: Enter the fully qualified path to the identity keystore:

   ```
   ORACLE_BASE/admin/domain_name/aserver/domain_
   name/certs/appIdentityKeyStore.jks
   ```

   Note: The example directory path given in this step is an example. Oracle does not recommend putting keystores into the `aserver` directory, but recommends putting the keystore is shared storage. Having a separate directory for certificates is a better solution.

   ---

   b. **Custom Identity Keystore Type**: Leave this field blank, it defaults to JKS.

   c. **Custom Identity Keystore Passphrase**: Enter the password *Keystore_Password* you provided in Section 8.3.2, "Creating an Identity Keystore Using the utils.ImportPrivateKey Utility."

   This attribute may be 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 reads only from the keystore, so whether or not you define this property depends on the requirements of the keystore.

7. In the **Trust** section, define properties for the trust keystore:

   a. **Custom Trust Keystore**: Enter the fully qualified path to the trust keystore:

   ```
   ORACLE_BASE/admin/domain_name/aserver/domain_
   name/certs/appTrustKeyStore.jks
   ```

   b. **Custom Trust Keystore Type**: Leave this field blank, it defaults to JKS.

   c. **Custom Trust Keystore Passphrase**: The password you provided in as *New_Password* in Section 8.3.3, "Creating a Trust Keystore Using the Keytool Utility."

      This attribute may be 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 reads only from the keystore, so whether or not you define this property depends on the requirements of the keystore.

8. Click **Save**.

9. To activate these changes, in the Change Center of the Administration Console, click **Activate Changes**.

10. Select **Configuration**, then **SSL**.

11. 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 8.3.2, "Creating an Identity Keystore Using the utils.ImportPrivateKey Utility."

12. Click **Save**.

13. Click **Activate Changes** in the Administration Console's Change Center to make the changes take effect.

14. Restart the server for which the changes have been applied.

Execute these steps for the Administration Server, the WLS_WSM*n* and the WLS_SOA*n* servers.

9 Server Migration

In this enterprise topology, you must configure server migration for the WLS_SOA1 and WLS_SOA2 managed servers. The WLS_SOA1 managed server is configured to restart on SOAHOST2 should a failure occur. The WLS_SOA2 managed server is configured to restart on SOAHOST1 should a failure occur. For this configuration, the WLS_SOA1 and WLS_SOA2 servers listen on specific floating IPs that are failed over by WLS Server Migration. Configuring server migration for the WLS_SOA*n* 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: Enabling Host Name Verification Certificates between SOAHOST1 and SOAHOST2 and the Administration Server
- Step 4: Editing the Node Manager's Properties File
- Step 5: Setting Environment and Superuser Privileges for the wlsifconfig.sh Script
- Step 6: Configuring Server Migration Targets
- Step 7: Testing the Server Migration

9.1 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:

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;
```

9.2 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. When you create 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/participation algorithm for the data source (do not choose the **Supports Global Transactions** option, or 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 SOA cluster
- Make sure the datasources' connection pool initial capacity is set to **0**. To do this, select **Services**, **JDBC**, and then **Datasources**. In the Datasources screen, click the **Datasource Name**, then click the **Connection Pool** tab, and enter **0** in the **Initial capacity** field.

Creating a Multi-Data Source

To create a multi-data source, complete these steps:

1. From Domain Structure window in the Oracle WebLogic Server Administration Console, expand the **Services** node, then click **Data Sources**.

   The Summary of JDBC Multi Data Source page appears.
2. Click **Datasources**, and then **Create New Multi Data Source**.
3. Click **Lock and Edit**.
4. Click **New**.

   The Create a New JDBC Multi Data Source page appears.
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 **SOA_Cluster** as the target.
10. Click **Next**.
11. Select **non-XA driver (the default)**.
12. Click **Next**.
13. Click **Create New Data Source**.
14. Enter *leasing-rac0* as name. Enter *jdbc/leasing-rac0* as JNDI name. Enter *oracle* as the database type. For the driver type, enter *Oracle Driver (Thin) for RAC Service-Instance connection Version 10,11.*

> **Note:** When creating the multi-datasources 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 port, and password for your leasing schema.
19. Click **Next**.
20. Click **Test Configuration** and verify the connection works.
21. Click **Next**.
22. Target the data source to SOA_Cluster.
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 SOA_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**.

9.3 Enabling Host Name Verification Certificates between SOAHOST1 and SOAHOST2 and the Administration Server

The third step is to create the appropriate certificates for host name verification between the Node Manager and the Administration Server. This procedure is described in Section 8.3, "Enabling Host Name Verification Certificates for Node Manager in SOAHOST1" and Section 8.5, "Enabling Host Name Verification Certificates for the Node Manager in SOAHOST2."

9.4 Editing the Node Manager's Properties File

The fourth step is to edit the Node Manager's properties file. This needs to be done for the node managers in both nodes where server migration is being configured. The nodemanager.properties files is located in the *WL_HOME*/common/nodemanager directory:

```
Interface=eth0
```

```
NetMask=255.255.255.0
```

```
UseMACBroadcast=true
```

- `Interface`

  This property specifies the interface name for the floating IP (for example, eth0).

> **Note:** Do not specify the sub interface, such as `eth0:1` or `eth0:2`. This interface is to be used without the `:0`, or `:1`. The 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`, or, `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 the 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 the 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 the shiphome to enable the Node Manager to start the managed servers.

2. Start the Node Manager on Node 1 and Node 2 by running the `startNodeManager.sh` script 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 SOAHOSTn, use the `Interface` environment variable as follows: `SOAHOSTn> export JAVA_OPTIONS=-DInterface=eth3` and start Node Manager after the variable has been set in the shell.

9.5 Setting Environment and Superuser Privileges for the wlsifconfig.sh Script

The fifth step is to set environment and superuser privileges for the `wlsifconfig.sh` script:

1. Ensure that your PATH environment variable includes these files:

Table 9–1 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, to set the environment and superuser privileges for the `wlsifconfig.sh` script, complete these steps:

     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.

9.6 Configuring Server Migration Targets

The sixth step is to configure server migration targets. Configuring Cluster Migration sets the `DataSourceForAutomaticMigration` property to true. Follow the steps below to configure cluster migration in a migration in a cluster:

1. Log into the Oracle WebLogic Server Administration Console (`http://<host>:<adminPort>/console`. Typically, `adminPort` is 7001 by default).
2. In the Domain Structure window, expand **Environment** and select **Clusters**. The Summary of Clusters page appears.
3. Click the cluster for which you want to configure migration (**SOA_Cluster**) in the Name column of the table.
4. Click the **Migration** tab.
5. Click **Lock and Edit**.
6. In the **Available** field, select the machine to which to allow migration and click the right arrow. In this case, select **SOAHOST1** and **SOAHOST2**.
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. Set the Candidate Machines for Server Migration. You must perform this task for all of the managed servers as follows:
    1. In Domain Structure window of the Oracle WebLogic Server Administration Console, expand **Environment** and select **Servers**.
    2. Select the server for which you want to configure migration.
    3. Click the **Migration** tab.
    4. 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_SOA1**, select **SOAHOST2**. For **WLS_SOA2**, select **SOAHOST1**.
    5. Select **Automatic Server Migration Enabled**. This enables the Node Manager to start a failed server on the target node automatically.
    6. Click **Save**.
    7. Click **Activate Changes**.
    8. Restart the Administration Server and the servers for which server migration has been configured

       To restart the Administration Server, use the procedure in Section 4.7, "Starting the Administration Server on SOAHOST1."

       Tip: Click **Customize this table** in the Summary of Servers page, 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.

9.7 Testing the Server Migration

The seventh and final step is to test the server migration. To verify that Server Migration is working properly, follow these steps:

From Node 1:

1. Stop the WLS_SOA1 managed server.

To do this, run this command:

```
SOAHOST1> kill -9 <pid>
```

pid specifies the process ID of the managed server. You can identify the pid in the node by running this command:

```
SOAHOST1> ps -ef | grep WLS_SOA1
```

2. Watch the Node Manager console: you should see a message indicating that WLS_SOA1's floating IP has been disabled.
3. Wait for the Node Manager to try a second restart of WLS_SOA1. Node Manager waits for a fence period of 30 seconds before trying this restart.
4. Once Node Manager restarts the server, stop it again. Now Node Manager should log a message indicating that the server will not be restarted again locally.

From Node2:

1. Watch the local Node Manager console. After 30 seconds since the last try to restart WLS_SOA1on Node 1, Node Manager on Node 2 should prompt that the floating IP for WLS_SOA1 is being brought up and that the server is being restarted in this node.
2. Access the soa-infra console in the same IP.

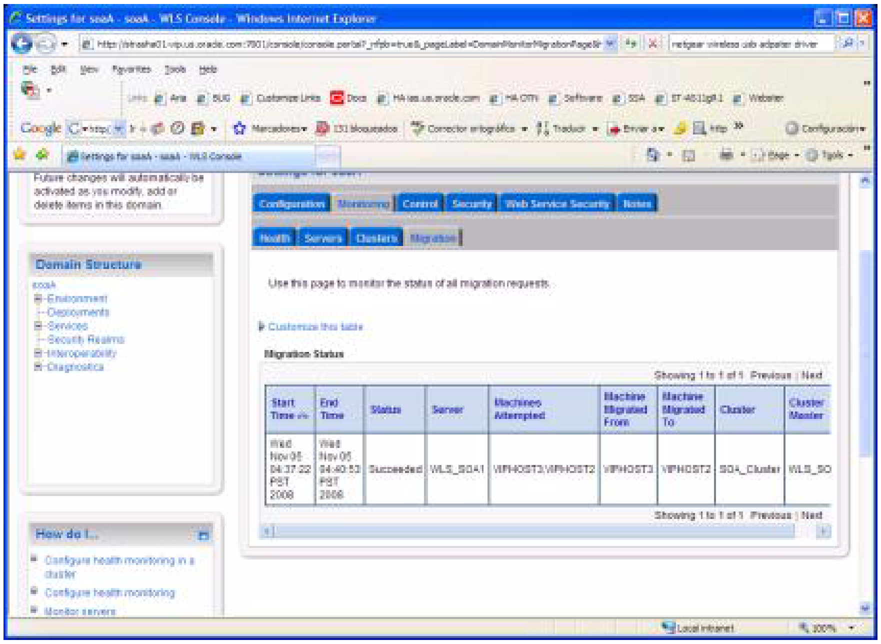
Verification From the Administration Console

Migration can also be verified in the Administration Console:

1. Log into the Administration Console.
2. Click on **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 9–1 Migration Status Screen in the Administration Console



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

Use this page to monitor the status of all migration requests.

Migration Status

Showing 1 to 1 of 1

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

Note: After a server is migrated, to fail it back to its original node/machine, stop the managed server from the Oracle WebLogic Administration Console and then start it again. The appropriate Node Manager starts the managed server on the machine to which it was originally assigned.

10

Managing the Topology

This chapter describes some operations that you can perform after you have set up the topology. These operations include monitoring, scaling, and backing up your topology.

This chapter contains the following sections:

10.1 Monitoring the Topology

For information on monitoring the topology, see chapters 7 and 8 of the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite*.

10.2 Deploying Composites and Artifacts in SOA Enterprise Deployment Topology

When deploying SOA composites to a SOA Enterprise Deployment Topology, deploy to a specific server's address and not to the load balancer address (soa.mycompany.com). Deploying to the load balancer address may require direct connection from the deployer nodes to the external load balancer address which may require additional ports to be opened in the firewalls used by the system.

For B2B, deploy agreements and purge/import metadata ONLY from the GUI available in B2B console instead of using the command line utility. Using the command line utility for these operations may cause inconsistencies and errors in the B2B system.

When deploying SOA Fusion Order Demo, the following additional steps are required in addition to the deployment steps provided in the FOD's README file).

1. Change the nostage property to **false** in the `build.xml` file of the Web applications so that ear files are copied to each node. Edit the `CreditCardAuthorization` and `OrderApprvalHumanTask` `build.xml` files,

located at FOD_dir\CreditCardAuthorization\bin and FOD_dir\OrderApprovalHumanTask\bin directories, and change the following field:

```
<target name="deploy-application">
     <wldeploy action="deploy" name="${war.name}"
       source="${deploy.ear.source}" library="false"
       nostage="false"
       user="${wls.user}" password="${wls.password}"
       verbose="false" adminurl="${wls.url}"
       remote="true" upload="true"
       targets="${server.targets}" />
   </target>
```

To:

```
<target name="deploy-application">
     <wldeploy action="deploy" name="${war.name}"
       source="${deploy.ear.source}" library="false"
       nostage="true"
       user="${wls.user}" password="${wls.password}"
       verbose="false" adminurl="${wls.url}"
       remote="true" upload="true"
       targets="${server.targets}" />
   </target>
```

2. Change the target for the Web applications so that deployments are targeted to the SOA Cluster and not to an individual server. Edit the build.properties file for FOD, located in the FOD_Dir/bin directory, and change the following field:

```
# wls target server (for shiphome set to server_soa, for ADRS use AdminServer)
server.targets=SOA_Cluster (the SOA cluster name in your SOA EDG)
```

3. Change the JMS seed templates so that instead of regular Destinations, Uniform Distributed Destinations are used and the JMS artifacts are targeted to the Enterprise Deployment JMS Modules. Edit the createJMSResources.seed file, located in the FOD_DIR\bin\templates directory, and change:

```
# lookup the SOAJMSModule - it's a system resource
     jmsSOASystemResource = lookup("SOAJMSModule","JMSSystemResource")

     jmsResource = jmsSOASystemResource.getJMSResource()

     cfbean = jmsResource.lookupConnectionFactory('DemoSupplierTopicCF')
     if cfbean is None:
         print "Creating DemoSupplierTopicCF connection factory"
         demoConnectionFactory =
 jmsResource.createConnectionFactory('DemoSupplierTopicCF')
         demoConnectionFactory.setJNDIName('jms/DemoSupplierTopicCF')
         demoConnectionFactory.setSubDeploymentName('SOASubDeployment')
 .
     topicbean = jmsResource.lookupTopic('DemoSupplierTopic')
     if topicbean is None:
         print "Creating DemoSupplierTopic jms topic"
         demoJMSTopic = jmsResource.createTopic("DemoSupplierTopic")
         demoJMSTopic.setJNDIName('jms/DemoSupplierTopic')
         demoJMSTopic.setSubDeploymentName('SOASubDeployment')
```

To:

```
jmsSOASystemResource = lookup("SOAJMSModule","JMSSystemResource")
```

```
jmsResource = jmsSOASystemResource.getJMSResource()

 topicbean=jmsResource.lookupTopic('DemoSupplierTopic_UDD')

if topicbean is None:
         print "Creating DemoSupplierTopicC jms topic"
         #create a udd - so clustering is automatically working and done
         demoJMSTopic =
jmsResource.createUniformDistributedTopic("DemoSupplierTopic_UDD")

         demoJMSTopic.setJNDIName('@jms.topic.jndi@')
         #Replace the subdeployment name with the one that appears in the WLS
AdminConsole as listed for the SOAJMSModule

         demoJMSTopic.setSubDeploymentName()

else: print "Found DemoSupplierTopic_UDD topic - noop"
```

10.3 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's 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 time out. 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 1000 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.

10.4 Configuring UMS Drivers

UMS driver configuration is not automatically propagated in a SOA or BAM cluster. This implies that users need to:

1. Apply the configuration of UMS drivers in each and every one of the servers in the Enterprise Deployment 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 WLS during deployment, for example, "3682yq").

In order to create the file in preparation for possible failovers, users can force a server migration and copy the file from the source node. For example, for BAM:

1. Configure the driver for WLS_BAM1 in BAMHOST1.
2. Force a failover of WLS_BAM1 to BAMHOST2. 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 BAMHOST1 to BAMHOST2:

   ```
   BAMHOST1> scp ORACLE_BASE/admin/<domain_name>/mserver/<domain_
   name>/servers/<server_name>/tmp/_WL_user/<ums_driver_
   name>/*/configuration/driverconfig.xml
   oracle@BAMHOST2:ORACLE_BASE/admin/<domain_name>/mserver/<domain_
   name>/servers/<server_name>/tmp/_WL_user/<ums_driver_name>/*/configuration/
   ```

It is required to restart the driver for these changes to take effect (that is, for the driver to consume the modified configuration). To restart the driver:

1. Log on to the Oracle WebLogic Administration console.
2. Expand the environment node on the navigation tree.

3. Click on **Deployments**.
4. Select the driver.
5. Click **Stop->When work completes** and confirm the operation.
6. Wait for the driver to transition to the "Prepared" state (refresh the administration console page, if required).
7. Select the driver again, and click **Start->Servicing all requests** and confirm the operation.

Make sure that you verify in Oracle Enterprise Manager Fusion Middleware Control that the properties for the driver have been preserved.

10.5 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 10.5.1, "Scaling Up the Topology (Adding Managed Servers to Existing Nodes)"
- Section 10.5.2, "Scaling Out the Topology (Adding Managed Servers to New Nodes)"

10.5.1 Scaling Up the Topology (Adding Managed Servers to Existing Nodes)

When you scale up the topology, you already have a node that runs a managed server that is configured with SOA components or a managed server with WSM-PM. The node contains a WebLogic Server home and an Oracle Fusion Middleware SOA home in shared storage. Use existing these installations (such as WebLogic Server home, Oracle Fusion Middleware home, and domain directories), when you create the new managed servers called WLS_SOA and WLS_WSM. You do not need to install WLS or SOA binaries at a new location or to run `pack` and `unpack`.

> **Note:** Because the BAM Server runs in active-passive, you cannot scale a BAM Server Managed Server. You can scale a BAM Web Applications server. The untargeting described in Section 7.6, "Untargeting the BAM Server System from WLS_BAM2" is required. The JMS configuration and the JMS configuration detailed in this section is not required.

1. Using the Oracle WebLogic Server Administration Console, clone WLS_SOA1 or WLS_WSM1 into a new managed server. The source managed server to clone should be one that already exists on the node where you want to run the new managed server.

   To clone a managed server, complete these steps:

   a. From the Domain Structure window of the Oracle WebLogic Server Administration Console, expand the **Environment** node and then **Servers**. The Summary of Servers page appears.

b. Click **Lock and Edit** and select the managed server that you want to clone (for example, WLS_SOA1).

c. Click **Clone**.

Name the new managed server WLS_SOA*n*, where *n* is a number that identifies the new managed server. In this case, assume that you are adding a new server to Node 1, where WLS_SOA1 was running.

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 as recommended for this server, enter 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. For WLS_WSM servers, run the Java Object Cache configuration utility again to include the new server in the JOC distributed cache as described in Section 4.17, "Configuring the Java Object Cache for Oracle WSM." You can use the same discover port for multiple WSM-PM servers in the same node. Repeat the steps provided in Section 4.17 for each WSM-PM server and the server list is updated.

4. Create JMS servers for SOA and UMS on the new managed server.

Note: These steps are not required for scaling up the WSM_PM managed server, only for WLS_SOA managed servers. They are not required either to scale up the BAM Web Applications system.

Create the JMS servers for SOA, UMS and BPM (if applicable) as follows:

a. Use the Oracle WebLogic Server Administration Console to create a new persistent store for the new SOAJMSServer (which will be created in a later step) and name it, for example, **SOAJMSFileStore_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/SOAJMSFileStore_N
```

Note: This directory must exist before the managed server is started or the start operation will fail.

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 UMS JMS server (which will be created in a later step) and name it, for example, **UMSJMSFileStore_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/UMSJMSFileStore_N
```

Note: This directory must exist before the managed server is started or the start operation will fail.

Note: It is also possible to 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. **For BPM Systems only**: Create a new persistence store for the new BPMJMSServer, for example, **BPMJMSFileStore_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/BPMJMSFileStore_N.

Note: This directory must exist before the managed server is started or the start operation fails.

You can also assign SOAJMSFileStore_N as store for the new BPM JMS Servers. For the purpose of clarity and isolation, individual persistent stores are used in the following steps.

f. **For BPM systems only**: Create a new JMS Server for BPM, for example, BPMJMSServer_N. Use the BPMJMSFileStore_N for this JMSServer. Target the BPMJMSServer_N Server to the recently created Managed Server (WLS_SOAn).

g. Target the UMSJMSSystemResource to the SOA_Cluster as it may have changed during extend operations. To do this, expand the **Services** node and then expand the **Messaging** node. Choose JMS Modules from the Domain Structure window of the Oracle WebLogic Server Administration Console. The JMS Modules page appears. Click UMSJMSSytemResource and open the Targets tab. Make sure all of the servers in the SOA_Cluster appear selected (including the recently cloned WLS_SOA*n*).

h. Update the SubDeployment Targets for SOA, UMS and BPM JMS Modules (if applicable) to include the recently created JMS servers.

   To do this, expand the **Services** node and then expand the **Messaging** node. Choose **JMS Modules** from the **Domain Structure** window of the Oracle WebLogic Server Administration Console. The JMS Modules page appears. Click on the JMS module (for SOA: SOAJMSModule, for BPM: BPMJMSMOdule and for UMS: UMSSYtemResource) represented as a hyperlink in the **Names** column of the table. The Settings page for module appears. Open the **SubDeployments** tab. The subdeployment for the deployment module appears.

Note: This subdeployment module name is a random name in the form of SOAJMSServerXXXXXX, UMSJMSServerXXXXXX, or BPMJMSServerXXXXXX, resulting from the Configuration Wizard JMS configuration for the first two servers (WLS_SOA1 and WLS_SOA2).

Click on it. Add the new JMS Server (for UMS add UMSJMSServer_N, for SOA add SOAJMSServer_N). Click **Save and Activate**.

5. Configuring Oracle Coherence for deploying composites for the new server as described in Section 5.4, "Configuring Oracle Coherence for Deploying Composites."

   Note: Only the **localhost** field must be changed for the server. Replace the localhost with the listen address of the new server added:

   ```
   Dtangosol.coherence.localhost=SOAHOST1VHNn
   ```

6. Configure the persistent store for the new server. This should be a location visible from other nodes as recommended in Section 2.3, "Shared Storage and Recommended Directory Structure."

   From the Administration Console, select the **Server_name** , and then 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.

7. Disable host name verification for the new managed server. Before starting and verifying the WLS_SOAN 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 the Node Manager in SOAHOST*n*.

   If the source server from which the new one has been cloned had already disabled hostname verification, these steps are not required (the hostname verification settings is propagated to the cloned server).

   To disable host name verification:

   a. In the Oracle Fusion Middleware 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 appears.

   d. Select **WLS_SOAn** in the **Names** column of the table.

      The Settings page for server appears.

   e. Click the **SSL** tab.

   f. Click **Advanced**.

   g. Set Hostname Verification to **None**.

   h. Click **Save**.

8. Configure server migration for the new managed server.

   Note: Because this is a scale-up operation, the node should already contain a Node Manager and environment configured for server migration that includes netmask, interface, wlsifconfig script superuser privileges, and so on. The floating IP for the new SOA managed server should also be already present.

To configure server migration using the Oracle WebLogic Server Administration Console, complete these steps:

a. In the Domain Structure window, expand the **Environment** node and then click **Servers**. The Summary of Servers page appears.

b. 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 appears.

c. Click the **Migration** subtab.

d. In the Migration Configuration section, select the servers that participate in migration in the Available window by clicking 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.

e. Choose the **Automatic Server Migration Enabled** option. This enables the Node Manager to start a failed server on the target node automatically.

f. Click **Save**.

g. Restart the Administration Server, managed servers, and Node Manager.

   To restart the Administration Server, use the procedure in Section 4.7, "Starting the Administration Server on SOAHOST1.".

9. Update the cluster address to include the new server:

   a. In the Administration Console, select **Environment**, and then **Cluster**.

   b. Click the **SOA_Cluster** server.

      The Settings screen for the SOA_Cluster appears.

   c. Click **Lock and Edit**.

   d. Add the new server's address and port to the **Cluster address** field. For example: **SOAHOST1VHN1:8011,SOAHOST2VHN1:8011,SOAHOST1VHN1 :8001**

   e. Save and activate the changes.

10. Test server migration for this new server. To test migration, perform the following from the node where you added the new server:

    a. Stop the WLS_SOAn 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 `ps -ef | grep WLS_SOAn`.

    b. Monitor the Node Manager Console for a message indicating that WLS_SOA1's floating IP has been disabled.

    c. Wait for the Node Manager to attempt a second restart of WLS_SOAn. Node Manager waits for a fence period of 30 seconds before trying this restart.

d. Once Node Manager restarts the server, stop it again. The Node Manager should log a message indicating that the server will not be restarted again locally.

10.5.2 Scaling Out the Topology (Adding Managed Servers to New Nodes)

When you scaling out the topology, you add new managed servers configured with SOA and or WSM-PM to new nodes.

Before performing the steps in this section, check that you meet these requirements:

Prerequisites

- There must be existing nodes running managed servers configured with SOA and WSM-PM within the topology
- The new node can access the existing home directories for WebLogic Server and SOA. (Use the existing installations in shared storage for creating a new WLS_SOA or WLS_WSM managed server. You do not need to install WebLogic Server or SOA 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.

To scale out the topology, complete these steps:

1. On the new node, mount the existing FMW Home, which should include the SOA installation and the domain directory, and ensure that the new node has access to this directory, just like the rest of the nodes in the domain.
2. To attach ORACLE_HOME in shared storage to the local Oracle Inventory, execute the following command:

```
SOAHOSTn>cd ORACLE_COMMON_HOME/oui/bin/attachHome.sh
SOAHOSTn>./attachHome.sh -jreLoc ORACLE_BASE/fmw/jrockit_160_<version>
```

   To update the Middleware home list, create (or edit, if another WebLogic installation exists in the node) the *$HOME/bea/beahomelist* file and add `MW_HOME` to it.

3. Log in to the Oracle WebLogic Administration Console.
4. Create a new machine for the new node that will be used, and add the machine to the domain.
5. Update the machine's Node Manager's address to map the IP of the node that is being used for scale out.
6. Use the Oracle WebLogic Server Administration Console to clone WLS_SOA1/WLS_WSM1 into a new managed server. Name it WLS_SOAn/WLS_WSM-PMn, where *n* is a number. Assign it to the new machine created above.

> **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. For WLS_WSM servers, run the Java Object Cache configuration utility again to include the new server in the JOC distributed cache as described in Section 4.17, "Configuring the Java Object Cache for Oracle WSM."

9. Create JMS Servers for SOA, BPM, (if applicable) and UMS on the new managed server.

   Note: These steps are not required for scaling out the WSM_PM managed server, only for WLS_SOA managed servers. They are not required either to scale up the BAM Web Applications system.

   Create the JMS servers for SOA and UMS as follows:

   a. Use the Oracle WebLogic Server Administration Console to create a new persistent store for the new SOAJMSServer (which will be created in a later step) and name it, for example, **SOAJMSFileStore_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/SOAJMSFileStore_N
      ```

      Note: This directory must exist before the managed server is started or the start operation will fail.

      ---

   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, and name it, for example, **UMSJMSFileStore_N**. As the directory for the persistent store, specify the path 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/UMSJMSFileStore _N
      ```

      Note: This directory must exist before the managed server is started or the start operation will fail.

      Note: It is also possible to 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. **For BPM Systems only**: Create a new persistence store for the new BPMJMSServer, for example, **BPMJMSFileStore_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/BPMJMSFileStore_N.

   > **Note:** This directory must exist before the managed server is started or the start operation fails.
   >
   > You can also assign SOAJMSFileStore_N as store for the new BPM JMS Servers. For the purpose of clarity and isolation, individual persistent stores are used in the following steps.

f. **For BPM systems only**: Create a new JMS Server for BPM, for example, BPMJMSServer_N. Use the BPMJMSFileStore_N for this JMSServer. Target the BPMJMSServer_N Server to the recently created Managed Server (WLS_SOAn).

g. Update the SubDeployment targets for the SOA JMS Module to include the recently created SOA JMS server. To do this, expand the **Services** node and then expand the **Messaging** node. Choose JMS Modules from the Domain Structure window of the Oracle WebLogic Server Administration Console. The JMS Modules page appears. Click SOAJMSModuleUDDs (represented as a hyperlink in the Names column of the table). The Settings page for SOAJMSModuleUDDs appears. Open the SubDeployments tab. The SOAJMSSubDM subdeployment appears.

   > **Note:** This subdeployment module results from updating the JMS configuration for the first two servers (WLS_SOA1 and WLS_SOA2) with the Uniform Distributed Destination Script (*soa-createUDD.py*), which is required for the initial Enterprise Deployment topology setup.

   Click on it. Add the new JMS server for SOA called SOAJMSServer_N to this subdeployment. Click **Save**.

h. Target the UMSJMSSystemResource to the SOA_Cluster as it may have changed during extend operations. To do this, expand the **Services** node and then expand the **Messaging** node. Choose JMS Modules from the Domain Structure window of the Oracle WebLogic Server Administration Console. The JMS Modules page appears. Click UMSJMSSytemResource and open the Targets tab. Make sure all of the servers in the SOA_Cluster appear selected (including the recently cloned WLS_SOA*n*).

i. Update the SubDeployment Targets for SOA, UMS and BPM JMS Modules (if applicable) to include the recently created JMS servers.

   To do this, expand the **Services** node and then expand the **Messaging** node. Choose **JMS Modules** from the **Domain Structure** window of the Oracle WebLogic Server Administration Console. The JMS Modules page appears. Click on the JMS module (for SOA: SOAJMSModule, for BPM: BPMJMSMOdule and for UMS: UMSSYtemResource) represented as a hyperlink in the **Names** column of the table. The Settings page for module appears. Open the **SubDeployments** tab. The subdeployment for the deployment module appears.

Note: This subdeployment module name is a random name in the form of SOAJMSServerXXXXXX, UMSJMSServerXXXXXX, or BPMJMSServerXXXXXX, resulting from the Configuration Wizard JMS configuration for the first two servers (WLS_SOA1 and WLS_SOA2).

Click on it. Add the new JMS Server (for UMS add UMSJMSServer_N, for SOA add SOAJMSServer_N). Click **Save and Activate**.

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

    ```
    SOAHOST1> cd ORACLE_COMMON_HOME/common/bin

    SOAHOST1> ./pack.sh -managed=true -domain=ORACLE_BASE/admin/domain_
    name/aserver/domain_name
    -template=soadomaintemplateScale.jar -template_name=soa_domain_templateScale
    ```

    Run the following command on SOAHOST1 to copy the template file created to SOAHOSTN

    ```
    SOAHOST1> scp soadomaintemplateScale.jar oracle@SOAHOSTN:/ ORACLE_COMMON_
    HOME/common/bin
    ```

    Run the `unpack` command on SOAHOSTN to unpack the template in the managed server domain directory as follows:

    ```
    SOAHOSTN> cd ORACLE_COMMON_HOME/common/bin

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

11. Configuring Oracle Coherence for deploying composites for the new server as described in Section 5.4, "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
    ```

12. Configure the persistent store for the new server. This should be a location visible from other nodes as recommended in Section 2.3, "Shared Storage and Recommended Directory Structure."

    From the Administration Console, select the **Server_name**, and then 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 starting and verifying 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 the Node Manager in SOAHOST*n*.

If the source server from which the new one has been cloned had already disabled hostname verification, these steps are not required (the hostname verification settings is propagated to the cloned server).

To disable host name verification:

a. In the Oracle Fusion Middleware 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 appears.

d. Select **WLS_SOAn** in the **Names** column of the table.

   The Settings page for server appears.

e. Click the **SSL** tab.

f. Click **Advanced**.

g. Set Hostname Verification to **None**.

h. Click **Save**.

14. Start Node Manager on the new node. To start Node Manager, use the installation in shared storage from the existing nodes, and start Node Manager by passing the host name of the new node as a parameter as follows:

```
SOAHOSTN> WL_HOME/server/bin/startNodeManager
```

15. Start and test the new managed server from the Oracle WebLogic Server Administration Console.

    a. Ensure that the newly created managed server, WLS_SOAn, is running.

    b. Access the application on the load balancer (`https://soa.mycompany.com/soa-infra`). The application should be functional.

Note: The HTTP Servers in the topology should round robin requests to the newly 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 takes place only if at least one of the servers listed in the WebLogicCluster directive is running.

16. Configure server migration for the new managed server.

Note: Because 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. The floating IP for the new SOA Managed Server is already present in the new node.

Log into the Oracle WebLogic Server Administration Console and configure server migration following these steps:

a. Expand the **Environment** node in the Domain Structure windows and then choose Servers. The Summary of Servers page appears.

b. Select the server (represented as hyperlink) for which you want to configure migration from the Names column of the table. The Setting page for that server appears.

c. Click the **Migration** tab.

d. In the Available field of the Migration Configuration section, click the right arrow to select the machines to which to allow migration.

> **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.

e. Select **Automatic Server Migration Enabled**. This enables the Node Manager to start a failed server on the target node automatically.

f. Click **Save**.

g. Restart the Administration Server, managed servers, and the Node Manager.

   To restart the Administration Server, use the procedure in Section 4.7, "Starting the Administration Server on SOAHOST1.".

17. Update the cluster address to include the new server:

    a. In the Administration Console, select **Environment**, and then **Cluster**.

    b. Click the **SOA_Cluster** server.

       The Settings screen for the SOA_Cluster appears.

    c. Click **Lock and Edit**.

    d. Add the new server's address and port to the **Cluster address** field. For example: **SOAHOST1VHN1:8011,SOAHOST2VHN1:8011,SOAHOSTNVHN1:8001**

    e. Save and activate the changes.

18. Test server migration for this new server. Follow these steps from the node where you added the new server:

    a. Abruptly stop the WLS_SOA*n* managed server by running `kill -9 <pid>` on the PID of the managed server. You can identify the PID of the node using `ps -ef | grep WLS_SOAn`.

    b. In the Node Manager Console you should see a message indicating that WLS_SOA1's floating IP has been disabled.

    c. Wait for the Node Manager to try a second restart of WLS_SOA*n*. Node Manager waits for a fence period of 30 seconds before trying this restart.

    d. Once Node Manager restarts the server, stop it again. Now Node Manager should log a message indicating that the server will not be restarted again locally.

10.6 Performing Backups and Recoveries

Table 10–1 lists the static artifacts to back up in the 11*g* SOA enterprise deployment.

Table 10–1 Static Artifacts to Back Up in the 11g SOA 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_HOME/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
Installation-related files		OraInventory, `<user_home>/bea/beahomelist`, `oraInst.loc`, oratab	N/A

Table 10–2 lists the runtime artifacts to back up in the 11*g* SOA enterprise deployment.

Table 10–2 Run-Time Artifacts to Back Up in the 11g SOA Enterprise Deployment

Type	Host	Location	Tier
DOMAIN HOME	SOAHOST1 and SOAHOST2	`ORACLE_BASE/admin/<domain_name>/mserver/<domain_name>/`	Application Tier
Application artifacts (EAR and WAR files)	SOAHOST1 and SOAHOST2	Find the application artifacts by viewing all of the deployments through administration console	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*.

> **Note:** ORACLE_HOME should be backed up if any changes are made to the XEngine configuration that are part of your B2B setup. These files are located under `ORACLE_HOME/soa/thirdparty/edifecs/XEngine`. To back up ORACLE_HOME, execute the following command:
>
> ```
> SOAHOST1> tar -cvpf fmwhomeback.tar MW_HOME
> ```

10.7 Troubleshooting

This section covers the following topics:

- Section 10.7.1, "Access to BAM Results in HTTP Error 404"
- Section 10.7.2, "Page Not Found When Accessing soa-infra Application Through Load Balancer"

- Section 10.7.3, "Error While Retrieving Oracle B2B Document Definitions"
- Section 10.7.4, "Soa-infra Application Fails to Start Due to Deployment Framework Issues (Coherence)"
- Section 10.7.5, "Incomplete Policy Migration After Failed Restart of SOA Server"
- Section 10.7.6, "SOA, BAM, or WMS Servers Fail to Start Due to Maximum Number of Processes Available in Database"
- Section 10.7.7, "Administration Server Fails to Start After a Manual Failover"
- Section 10.7.8, "Error While Activating Changes in Administration Console"
- Section 10.7.9, "SOA/BAM Server Not Failed Over After Server Migration"
- Section 10.7.10, "SOA/BAM Server Not Reachable From Browser After Server Migration"
- Section 10.7.11, "SOA Server Stops Responding after Being Active and Stressed for a Period of Time."
- Section 10.7.12, "Exceptions While Performing Deploy/Purge/Import Operations in the B2B Console."
- Section 10.7.13, "OAM Configuration Tool Does Not Remove URLs"
- Section 10.7.14, "Redirecting of Users to Login Screen After Activating Changes in Administration Console"
- Section 10.7.15, "Redirecting of Users to Administration Console's Home Page After Activating Changes to OAM"
- Section 10.7.16, "Configured JOC Port Already in Use"

10.7.1 Access to BAM Results in HTTP Error 404

If accessing the BAM application results in the HTTP 404 error ("Not Found"), a probable cause is that the BAM server was started before the start of the database instance where BAM schemas reside. In this case, shut down the BAM instance and restart it after ensuring that the database is already up.

10.7.2 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.

10.7.3 Error While Retrieving Oracle B2B Document Definitions

Problem: Error happens when trying to retrieve a document definition XSD from Oracle B2B. B2B resides in a cluster and is accessed through a load balancer. B2B console report the following:

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

Solution: This occurs if you do not set the frontend HTTP host and port for the Oracle WebLogic cluster where Oracle B2B resides. To eliminate this error, set the front end address for the SOA Cluster:

1. In the WebLogic Server Administration Console, in the Change Center section, click **Lock & Edit**.
2. In the left pane, choose the **Environment in the Domain Structure** window and then choose **Clusters**. The Summary of Clusters page appears.
3. Select the `WLS_SOA` cluster.
4. Select **HTTP**.
5. Set the values for the following:
   - **Frontend Host:** `soa.mycompany.com`
   - **Frontend HTTPS Port:** `443`
   - **Frontend HTTP Port:** `80`
6. Click **Save**.
7. To activate the changes, click **Activate Changes** in the Change Center section of the Administration Console.
8. Restart the servers to make the Frontend Host directive in the cluster effective.

10.7.4 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 runtime 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:

```
oracle.integration.platform.blocks.deploy.CompositeDeploymentCoordinatorMessage
s errorUnableToStartCoherence
```

are reported in the server's output log if any of the addresses is not resolved correctly.

10.7.5 Incomplete Policy Migration After Failed Restart of SOA Server

Problem: The SOA server fails to start through the administration console *before* setting the 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>
```

10.7.6 SOA, BAM, or WMS Servers Fail to Start Due to Maximum Number of Processes Available in Database

Problem: SOA, WSM or BAM Server fails to start. The domain has been extended for new types of managed server (for example, SOA extended for BAM) or the system has been scaled up (added new servers of the same type). The SOA/BAM or WSM 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.

10.7.7 Administration Server Fails to Start After a Manual Failover

Problem: Administration Server fails to start after the Administration Server node failed and manual failover to another nodes is performed. The Administration Server output log reports the following:

```
<Feb 19, 2009 3:43:05 AM PST> <Warning> <EmbeddedLDAP> <BEA-171520> <Could not
obtain an exclusive lock for directory: ORACLE_BASE/admin/soadomain/aserver/
soadomain/servers/AdminServer/data/ldap/ldapfiles. Waiting for 10 seconds and then
retrying in case existing WebLogic Server is still shutting down.>
```

Solution: When restoring a node after a node crash and using shared storage for the domain directory, you may see this error in the log for the Administration Server due to unsuccessful lock cleanup. To resolve this error, remove the file *ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>/servers/AdminServer/data/ldap/ldapfiles/EmbeddedLDAP.lok*.

10.7.8 Error While Activating Changes in Administration Console

Problem: Activation of changes in Administration Console fails after changes to a server's start configuration have been performed. The Administration Console reports the following when clicking "Activate Changes":

```
An error occurred during activation of changes, please see the log for details.
 [Management:141190]The commit phase of the configuration update failed with an
exception:
In production mode, it's not allowed to set a clear text value to the property:
PasswordEncrypted of ServerStartMBean
```

Solution: This may happen when start parameters are changed for a server in the Administration Console. In this case, either provide username/password information in the server start configuration in the Administration Console for the specific server whose configuration was being changed, or remove the `<password-encrypted></password-encrypted>` entry in the *config.xml* file (this requires a restart of the Administration Server).

10.7.9 SOA/BAM 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 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.

10.7.10 SOA/BAM Server Not Reachable From Browser After Server Migration

Problem: Server migration is working (SOA/BAM Server is restarted in the failed over node) but the *<Virtual Hostname>:8001/soa-infra* URL is not reachable 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 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.

10.7.11 SOA Server Stops Responding after Being Active and Stressed for a Period of Time

Problem: WLS_SOA starts properly and functions for a period of time, but becomes unresponsive after running an application that uses the Oracle File Adapter or Oracle FTP Adapter. The log file for the server reports the following:

```
<Error> <Server> <BEA-002606> <Unable to create
a server socket for listening on channel "Default". The address
X.X.X.X might be incorrect or another process is using port 8001:
@ java.net.SocketException: Too many open files.>
```

Solution: For composites with Oracle File and FTP Adapters, which are designed to consume a very large number of concurrent messages, set the number of open files parameter for your operating system to a greater value. For example, to set the number of open files parameter to 8192 for Linux, use the `ulimit -n 8192` command. The value must be adjusted based on the expected system's load.

10.7.12 Exceptions While Performing Deploy/Purge/Import Operations in the B2B Console

Problem: Deployment of new agreements or purging/importing new metadata fails, and the output logs for the SWLS_SOA server reports "[java] MDS-02202: Content of the metadata object" for deployment or "postTransfer: MDS-00521: error while reading document..." for purge/import.

Solution: This is caused by timing and load balancing mechanism in the operation. The exceptions are unlikely to happen, so a retry of the operation will typically succeed. There is no cleanup or any other additional steps required.

10.7.13 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.

10.7.14 Redirecting of Users to Login Screen After Activating Changes in Administration Console

Problem: After configuring OHS and load balancer to access the Oracle WebLogic Administration Console, some activation changes cause the redirection to the login screen for the admin console.

Solution: This is the result of the console attempting to follow changes to port, channel, and security settings as a user makes these changes. For certain changes, the console may redirect to the Administration Server's listen address. Activation is completed regardless of the redirection. It is not required to log in again; users can simply update the URL to `soa.mycompany.com/console/console.portal` and directly access the home page for the Administration Console.

> **Note:** This problem will not occur if you have disabled tracking of the changes described in this section.

10.7.15 Redirecting of Users to Administration Console's Home Page After Activating Changes to OAM

Problem: After configuring OAM, some activation changes cause the redirection to the Administration Console's home page (instead of the context menu where the activation was performed).

Solution: This is expected when OAM SSO is configured and is the result of the redirections performed by the Administration Server. Activation is completed regardless of the redirection. If required, users may "manually" navigate again to the desired context menu.

10.7.16 Configured JOC Port Already in Use

Problem: Attempts to start a Managed Server that uses the Java Object Cache, such as OWSM or WebCenter Spaces Managed Servers, fail. The following errors appear in the logs:

```
J2EE JOC-058 distributed cache initialization failure
J2EE JOC-043 base exception:
J2EE JOC-803 unexpected EOF during read.
```

Solution: Another process is using the same port that JOC is attempting to obtain. Either stop that process, or reconfigure JOC for this cluster to use another port in the recommended port range.

10.7.17 SOA or BAM Server Fails to Start

The SOA or BAM server fails to start for the first time and reports parsing failure in config.xml.

Problem: A server that is being started for the first time using Node Manager fails to start. A message such as the following appears in the server's output log:

```
<Critical> <WebLogicServer> <eicfdcn35> <wls_server1> <main> <<WLS Kernel>> <> <>
<1263329692528> <BEA-000386> <Server subsystem failed. Reason:
weblogic.security.SecurityInitializationException: Authentication denied: Boot
identity not valid; The user name and/or password from the boot identity file
(boot.properties) is not valid. The boot identity may have been changed since the
boot identity file was created. Please edit and update the boot identity file with
the proper values of username and password. The first time the updated boot
identity file is used to start the server, these new values are encrypted.
```

The Managed Server is trying to start for the first time, in MSI (managed server independence) mode. The Server has not been able to retrieve the appropriate configuration for the first start. The Managed Server must be able to communicate with the Administration Server on its first startup.

Solution: Make sure communication between the Administration Server´s listen address and the Managed Server´s listen address is possible (ping the Administration Server's listen address from the Managed Server's node, and telnet to the Administration Server's listen address and port). Once communication is enabled, pack and unpack the domain again to the new node or (if other servers are already running correctly in the same domain directory), delete the *OARCLE_BASE*/admin/*domain_name*/mserver/*domain_name*/servers/*server_name*/data/nodemanager/ directory and restart the server.

10.8 Best Practices

This section covers the following topics:

- Section 10.8.1, "Preventing Timeouts for SQLNet Connections"
- Section 10.8.2, "Auditing"

10.8.1 Preventing Timeouts for SQLNet Connections

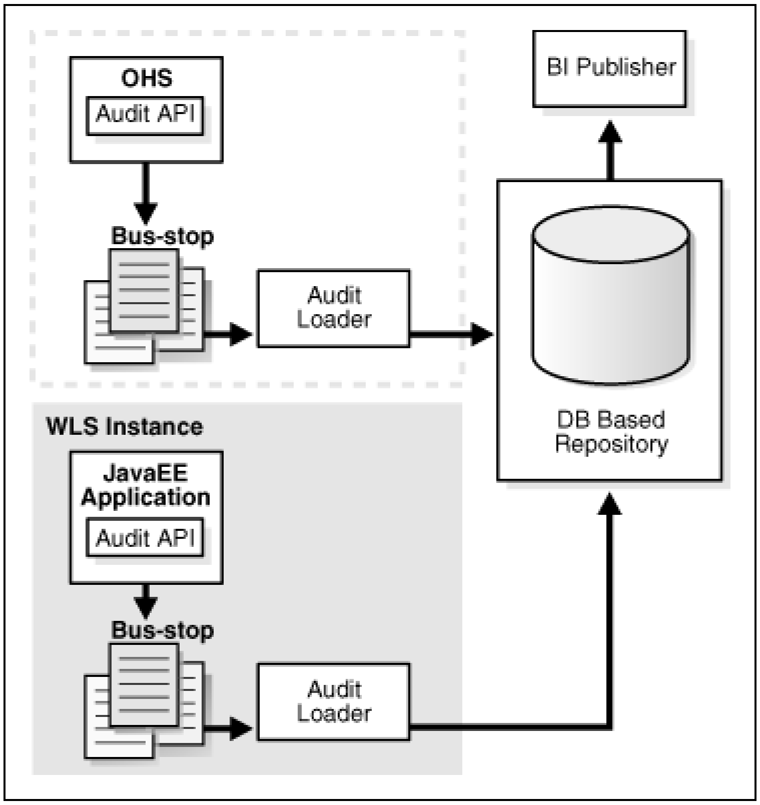
Much of the Enterprise Deployment production deployment involves firewalls. Because database connections are made across firewalls, Oracle recommends that the firewall be configured so that the database connection is not timed out. For Oracle Real Application Clusters (Oracle RAC), the database connections 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.

10.8.2 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 10–1 is a high-level architectural diagram of the Oracle Fusion Middleware Audit Framework.

Figure 10–1 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 runtime, applications may call these APIs where appropriate to audit the necessary information about a particular event happening in the application code. The interface allows applications to specify event details such as username and other attributes needed to provide the context of the event being audited.

- **Audit Events and Configuration**

The Oracle Fusion Middleware Audit Framework provides a set of generic events for convenient mapping to application audit events. Some of these include common events such as authentication. The framework also allows applications to define application-specific events.

These event definitions and configurations are implemented as part of the audit service in Oracle Platform Security Services. Configurations can be updated through Enterprise Manager (UI) and WLST (command-line tool).

- **Audit Bus-stop**

  Bus-stops are local files containing audit data before they are pushed to the audit repository. In the event where no database repository is configured, these bus-stop files can be used as a file-based audit repository. The bus-stop files are simple text files that can be queried easily to look up specific audit events. When a DB-based repository is in place, the bus-stop acts as an intermediary between the component and the audit repository. The local files are periodically uploaded to the audit repository based on a configurable time interval.

- **Audit Loader**

  As the name implies, audit loader loads the files from the audit bus-stop into the audit repository. In the case of platform and JavaEE application audit, the audit loader is started as part of the JavaEE container start-up. In the case of system components, the audit loader is a periodically spawned process.

- **Audit Repository**

  Audit Repository contains a pre-defined Oracle Fusion Middleware Audit Framework schema, created by Repository Creation Utility (RCU). Once configured, all the audit loaders are aware of the repository and upload data to it periodically. The audit data in the audit repository is expected to be cumulative and will grow overtime. Ideally, this should not be an operational database used by any other applications - rather, it should be a standalone RDBMS used for audit purposes only. In a highly available configuration, Oracle recommends that you use an Oracle Real Application Clusters (Oracle RAC) database as the audit data store.

- **Oracle Business Intelligence Publisher**

  The data in the audit repository is exposed through pre-defined reports in Oracle Business Intelligence Publisher. The reports allow users to drill down the audit data based on various criteria. For example:

  - Username
  - Time Range
  - Application Type
  - Execution Context Identifier (ECID)

For more introductory information for the Oracle Fusion Middleware Audit Framework, see the "Introduction to Oracle Fusion Middleware Audit Framework" chapter in the *Oracle Fusion Middleware Security Guide*.

For information on how to configure the repository for Oracle Fusion Middleware Audit Framework, see the "Configuring and Managing Auditing" chapter in the *Oracle Fusion Middleware Security Guide*.

The Enterprise Deployment 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.

11

Integration With Oracle Identity Management

This chapter describes how to integrate Oracle SOA Suite with Oracle Identity Management. It contains the following sections:

11.1 Credential and Policy Store Configuration

The following topics describe credential and policy store configuration in detail:

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 Oracle Fusion Middleware SOA Suite Enterprise Deployment topology uses different domain homes for the Administration Server and the Managed Server, thus Oracle requires the use of an LDAP store as policy and credential store for integrity and consistency. By default Oracle WebLogic Server domains use an XML file for the policy store. The following sections describe the steps required to change the default store to Oracle Internet Directory LDAP for credentials or policies.

Note: The backend repository for the policy store and the credential store must use the same kind of LDAP server. To preserve this coherence, note that reassociating one store implies reassociating the other one, that is, the re-association of both the credential and the policy stores is accomplished as a unit using Enterprise Manager 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. In this section, steps are provided to configure Oracle Internet Directory LDAP as a credential store for the Oracle Fusion Middleware SOA 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 in the *ORACLE_BASE*/admin/*domain_name*/aserver/*domain_name*/servers/AdminServer/security directory for the Administration Server.

To configure the credential store to use LDAP, set the proper authenticator using the WebLogic Server Console:

1. Log in to the WebLogic Server Console.
2. Click the **Security Realms** link on the left navigational bar.
3. Click the **myrealm** default realm entry to configure it.
4. Open the **Providers** tab within the realm.
5. Observe that there is a `DefaultAuthenticator` provider configured for the realm.
6. Click **Lock and 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:

1. OID LDAP Authenticator: **SUFFICIENT**
2. 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 the Oracle Fusion Middleware SOA Suite Enterprise Deployment WebLogic Domain. 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 SOA Enterprise Deployment WebLogic domain will be provisioned with the DNs below:

- Admin User DN:

  ```
  cn=weblogic_soa,cn=Users,dc=us,dc=mycompany,dc=com
  ```

- Admin Group DN:

  ```
  cn=SOA Administrators,cn=Groups,dc=us,dc=mycompany,dc=com
  ```

Follow these steps to provision the admin user and admin group in Oracle Internet Directory:

1. Create an ldif file named `admin_user.ldif` with the contents shown below and then save the file:

   ```
   dn: cn=weblogic_soa, cn=Users, dc=us, dc=mycompany, dc=com
   orclsamaccountname: weblogic_soa
   givenname: weblogic_soa
   sn: weblogic_soa
   userpassword: Welcome1
   obver: 10.1.4.0
   mail: weblogic_soa
   objectclass: top
   objectclass: person
   objectclass: organizationalPerson
   objectclass: inetorgperson
   objectclass: orcluser
   objectclass: orcluserV2
   objectclass: oblixorgperson
   uid: weblogic_soa
   cn: weblogic_soa
   description: Admin User for the SOA 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. The ORACLE_HOME environment variable must be set for the `ldapadd` command to succeed.

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=SOA Administrators, cn=Groups, dc=us, dc=mycompany, dc=com
displayname: SOA Administrators
objectclass: top
objectclass: groupOfUniqueNames
objectclass: orclGroup
uniquemember: cn=weblogic_soa,cn=users,dc=us,dc=mycompany,dc=com
cn: SOA Administrators
description: Administrators Group for the SOA Domain
```

4. Run the `ldapadd` command located under the ORACLE_HOME/bin/ directory to provision the group in Oracle Internet Directory (the command is shown as two lines in the example below for readability purposes, but you should enter the command on a single line):

```
OIDHOST1> ORACLE_HOME/bin/ldapadd -h oid.mycompany.com -p 389 -D
cn="orcladmin" -w welcome1 -c -v -f admin_group.ldif
```

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 into the WebLogic Administration Server Console.
2. In the left pane of the console, click **Security Realms**.
3. On the Summary of Security Realms page, click **myrealm** under the Realms table.
4. On the Settings page for myrealm, click the **Roles & Policies** tab.
5. On the Realm Roles page, expand the **Global Roles** entry under the **Roles** table. This brings up the entry for **Roles**. Click on the **Roles** link to bring up the Global Roles page.
6. On the Global Roles page, click the **Admin** role to bring up the Edit Global Role page:
    a. On the Edit Global Roles page, under the **Role Conditions** table, click the **Add Conditions** button.
    b. On the Choose a Predicate page, select **Group** from the drop down list for predicates and click **Next**.
    c. On the Edit Arguments Page, specify **SOA 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 SOA Administrators Group as an entry.
9. Click **Save** to finish adding the Admin Role to the SOA 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_soa user.

> **Note:** Each SOA application has its own predefined roles and groups defined for administration and monitoring. By default, the "Administrator" group allows these operations. However, the "Administrator" group may be too broad. For example, you may not want B2B Administrators to be WebLogic Server Domain Administrators where SOA is running. Therefore, you may wish to create a a more specific group, such as "SOA Administrators." In order for the different applications to allow the SOA Administrator group to administer the different systems, you must add the required roles to the SOA Administrator group. For example, for B2B's Administration, add the B2BAdmin role to the SOA Administrators group, for 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/domainName/aserver/domainName/servers/
AdminServer/security
```

2. Rename the existing `boot.properties` file:

```
SOAHOST1> mv boot.properties boot.properties.backup
```

3. Use a text editor to create a file called `boot.properties` under the security directory. Enter the following lines in the file:

```
username=weblogic_soa
password=welcome1
```

4. Save the file.

5. Stop the Administration Server:

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

6. Start the Administrator Server using the procedure in Section 4.7, "Starting the Administration Server on SOAHOST1."

11.1.2.3 Reassociating the Domain Credential Store

The re-association 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

11.1.3 Policy Store Configuration

The domain policy store is the repository of system and application-specific policies. In a given domain, there is one store that stores all policies that all applications deployed in the domain may use. This section provides the steps to configure Oracle Internet Directory LDAP as the policy store for the Oracle Fusion Middleware SOA 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. *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_soa
   cn: jpsroot_soa
   objectclass: top
   objectclass: OrclContainer
   ```

   The distinguished name of the root node (illustrated by the string `jpsroot_soa` above) must be distinct from any other distinguished name. One root node can be shared by multiple WebLogic domains. It is not required that this node be created at the top level, as long as read and write access to the subtree is granted to the Oracle Internet Directory administrator.

2. Import this data into Oracle Internet Directory server using the command `ldapadd`, as illustrated in the following example (the command is shown as two lines in the example below for readability purposes, but you should enter the command on a single line):

   ```
   OIDHOST1> ORACLE_HOME/bin/ldapadd -h ldap_host -p ldap_port -D
   cn=orcladmin -w password -c -v -f jpstestnode.ldif
   ```

3. Verify that the node has been successfully inserted using the command `ldapsearch`, as illustrated in the following example (the command is shown as two lines in the example below for readability purposes, but you should enter the command on a single line):

   ```
   OIDHOST1> ORACLE_HOME/bin/ldapsearch -h ldap_host -p ldap_port -D
   cn=orcladmin -w password -b "cn=jpsroot_soa" objectclass="orclContainer"
   ```

4. When using Oracle internet Directory as the LDAP-Based Policy Store run the utility `oidstats.sql` in the INFRADBHOSTs to generate database statistics for optimal database performance:

   ```
   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- or LDAP-based repository to an LDAP-based repository, that is, re-association changes the repository preserving the integrity of the data stored. For each policy in the source policy store, re-association searches the target LDAP directory and, if it finds a match, it updates the matching policy as appropriate. If none is found, it simply migrates the policy as is.

At any time, after a domain policy store has been instantiated, a file- or LDAP-based policy store can be reassociated into an LDAP-based policy store storing the same data. To support it, the domain has to be configured, as appropriate, to use an LDAP policy store.

The re-association 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_COMMONHOME/common/bin
SOAHOST1>./wlst.sh
```

2. Connect to the WebLogic Administration Server using the `wlst connect` command shown below:

   Syntax:

```
connect('AdminUser',"AdminUserPassword",t3://hostname:port)
```

   For example:

```
connect("weblogic","welcome1","t3://ADMINVHN:7001")
```

3. Run the `reassociateSecurityStore` command as shown below:

   Syntax:

```
reassociateSecurityStore(domain="domainName",admin="cn=orcladmin",
password="orclPassword",ldapurl="ldap://LDAPHOST:LDAPPORT",servertype="OID",
jpsroot="cn=jpsroot_soa")
```

   For example:

```
wls:/SOAEDGDomain/serverConfig>reassociateSecurityStore(domain="soaedg_domain",
admin="cn=orcladmin",password="welcome1",ldapurl="ldap://oid.mycompany.com:389"
,servertype="OID",jpsroot="cn=jpsroot_soa")
```

   The output for the command is shown below:

```
{servertype=OID,jpsroot_soa=cn=jpsroot_soa_idm_idmhost1,admin=cn=orcladmin,
domain=IDMDomain,ldapurl=ldap://oid.mycompany.com:389,password=welcome1}
Location changed to domainRuntime tree. This is a read-only tree with
```

```
DomainMBean as the root.

For more help, use help(domainRuntime)

Starting Policy Store reassociation.
LDAP server and  ServiceConfigurator setup done.

Schema is seeded into LDAP server
Data is migrated to LDAP server
Service in LDAP server after migration has been tested to be available
Update of jps configuration is done
Policy Store reassociation done.
Starting credential Store reassociation
LDAP server and  ServiceConfigurator setup done.
Schema is seeded into LDAP server
Data is migrated to LDAP server
Service in LDAP server after migration has been tested to be available
Update of jps configuration is done
Credential Store reassociation done
Jps Configuration has been changed. Please restart the server.
```

4. Restart the Administration Server after the command completes successfully.

   To restart the Administration Server, use the procedure in Section 4.7, "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 10g Integration

This section describes how to set up Oracle Access Manager 10g as the single sign-on solution for the Oracle SOA Suite Enterprise Deployment topology.

This section contains the following Topics:

- Section 11.2.1, "Overview of Oracle Access Manager Integration"
- Section 11.2.2, "Prerequisites for Oracle Access Manager"
- Section 11.2.3, "Using the OAM Configuration Tool"
- Section 11.2.4, "Installing and Configuring WebGate"
- Section 11.2.5, "Changing the CacheControl Headers in the SOA_EDG_AG for Oracle BAM"
- Section 11.2.6, "Configuring IP Validation for the Webgate"
- Section 11.2.7, "Setting Up WebLogic Authenticators"

11.2.1 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 chapter explains the procedure for configuring the SOA installation with an existing OAM installation and the underlying directory service. Oracle recommends using either Oracle Internet Directory (OID) or Oracle Virtual Directory (OVD) or both of these directory services.

> **Note:** The SOA Enterprise Deployment topology described in this book uses a Single Sign-On configuration where both the SOA System and the Single Sign-On System are in the same network domain (mycompany.com) For a multi-domain configuration, please refer to the required configuration steps in "Chapter 7, Configuring Single Sign-On," of the *Oracle Access Manager Access Administration Guide*.

11.2.2 Prerequisites for Oracle Access Manager

The setup for Oracle Access Manager (OAM) assumes an existing OAM installation complete with Access Managers and a policy protecting the Policy Manager. For more information on installing and configuring an OAM installation, see *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*. This setup includes a directory service such as Oracle Internet Directory (OID) either as a stand-alone or as part of an Oracle Virtual Directory (OVD) configuration. This chapter will provide the necessary steps for configuring your SOA installation with either OID or OVD.

In addition, the OAM installation should have its own Web server configured with WebGate. This section also provides the steps for using the OAM Web server as a delegated authentication server.

11.2.3 Using the OAM Configuration Tool

The OAM Configuration Tool (oamcfg) starts a series of scripts and setup the required policies. It requires various parameters as inputs. Specifically, it creates the following:

1. A Form Authentication scheme in OAM

2. Policies to enable authentication in WebLogic Server
3. A WebGate entry in OAM to enable Oracle HTTP Server WebGates (from your Web Tier) to protect your configured application
4. A Host Identifier, depending on the scenario chosen (a default host identifier would be used, if not provided) Policies to protect and unprotect application specific URLs.

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, "SOA Composite Applications and Oracle Access Manager Logout Guidelines"
- Section 11.2.3.4, "Verifying Successful Creation of the Policy Domain and AccessGate"
- Section 11.2.3.5, "Updating the Host Identifier"
- Section 11.2.3.6, "Updating the WebGate Profile"
- Section 11.2.3.7, "Adding Additional Access Servers"
- Section 11.2.3.8, "Configuring Delegated Form Authentication"

11.2.3.1 Collecting the Information for the OAM Configuration Tool

The following information should be collected or prepared prior to running the OAM Configuration tool:

1. **Password:** Create a secure password. This will be used as the password for the WebGate installation created later.
2. **LDAP Host:** host name of the Directory Server or Load Balancer address in the case of an HA/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_aa_host:** host name of an Oracle Access Manager
7. **oam_aa_port:** port of the Oracle Access Manager

11.2.3.2 Running the OAM Configuration Tool

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 SOAHOST1.

The OAM Configuration Tool provides a way to register protected and public resources into the OAM system. The list of protected resources to be added to the OAM system is as follows:

```
/integration/worklistapp
/integration/worklistapp/.../*
/workflow/sdpmessagingsca-ui-worklist
/workflow/sdpmessagingsca-ui-worklist/.../*
/b2bconsole
```

```
/b2bconsole/.../*
/sdpmessaging/userprefs-ui
/sdpmessaging/userprefs-ui/…/*
/DefaultToDoTaskFlow
/DefaultToDoTaskFlow/…/*
/em
/em/.../*
/console
/console/.../*
/soa/composer
/soa/composer/.../
/OracleBAM (For BAM systems only)
/OracleBAM/…/* (For BAM systems only)
/bpm/composer (For BPM systems only)
/bpm/composer/…/* (For BPM systems only)
/bpm/workspace
/bpm/workspace/…/*
/soa-infra
/soa-infra/deployer
/soa-infra/events/edn-db-log
/soa-infra/cluster/info
/inspection.wsil/
```

The list of public resources is as follows:

```
/soa-infra/services/.../*
/soa-infra/directWSDL
/soa-infra/directWSDL/.../*
/OracleBAMWS
/OracleBAMWS/.../*
/ucs/messaging/webservice
/ucs/messaging/webservice/.../*
```

Where "/…/*" implies all resources under the base url context.

The OAM Configuration Tool should be run as follows for OAM 10g registration (all on a single command line):

```
MW_HOME/jrockit_160_<version>/bin/java -jar oamcfgtool.jar mode=CREATE
app_domain="SOA_EDG"
protected_uris="$URI_LIST"
app_agent_password=<Password_to_be_provisioned_for_App_Agent>
ldap_host=OID.MYCOMPANY.COM
ldap_port=389
ldap_userdn="cn=orcladmin"
ldap_userpassword=<Password_of_LDAP_Admin_User>
oam_aaa_host=OAMHOST1
oam_aaa_port=OAMPORT1
```

Define the `$URI_LIST` variable to contain the list of URIs you want to protect as follows:

```
/integration/worklistapp
/workflow/sdpmessagingsca-ui-worklist
/b2bconsole
/sdpmessaging/userprefs-ui
/DefaultToDoTaskFlow
/em
/console
/soa/composer
/OracleBAM <!-- (For BAM systems only) -->
/bpm/composer <!-- (For BPM systems only) -->
```

```
/bpm/workspace <!-- (For BPM systems only) -->
/soa-infra
/soa-infra/deployer
/soa-infra/events/edn-db-log
/soa-infra/cluster/info
/inspection.wsil
```

> **Note:** If BAM is installed later or other additional URLs need to be protected, the OAM configuration tool should be executed again using the same `app_domain` and including *all* the URLs that would be protected (not just the new ones).

Define the $PUBLIC_URI_LIST variable to contain the list of URIs you want to set as not protected/public as follows:

```
/soa-infra/services/.../*
/soa-infra/directWSDL
/soa-infra/directWSDL/.../*
/OracleBAMWS
/OracleBAMWS/.../*
/ucs/messaging/webservice
/ucs/messaging/webservice/.../*
```

If your command ran successfully, you should see the following output:

```
Processed input parameters
Initialized Global Configuration
Successfully completed the Create operation
Operation Summary:
Policy Domain: SOA_EDG
Host Identifier: SOA_EDG
Access Gate ID: SOA_EDG_AG
```

11.2.3.3 SOA Composite Applications and Oracle Access Manager Logout Guidelines

For a SOA composite application complying with Oracle Access Manager logout guidelines (in particular, a composite that invokes a logout through `/adfAuthentication?logout=true&end_url=<someURI>`), integrating the composite into an Oracle Access Manager 10g 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 10g WebGate does not process end_url.

For information about configuration procedures, see *Oracle Fusion Middleware Security Guide*.

11.2.3.4 Verifying Successful Creation of the Policy Domain and AccessGate

Verifying the Policy Domain

To verify the policy domain, complete these steps:

1. Log on to the Oracle Access Manager: `http://OAMADMINHOST:<port>/access/oblix/`

2. Click **Policy Manager**.

3. Click the **My Policy Domains** link on the left panel, you will see a list of all policy domains, among which the domain you just created will be listed. It will have the suffix _PD (for example, `SOA_EDG_PD`). In the third column (URL prefixes, you will also see the URIs you specified during the creation of this domain).
4. Click the link to the policy domain you just created. you will land in the General area of this domain.
5. Click the **Resources** tab, you will see the URIs you specified. You can also click other tabs to view other settings.

Verifying the AccessGate Configuration

To verify the AccessGate configuration, complete these steps:

1. Click the **Access System Console** link on the top right hand side (this acts like a toggle; after you click it, it becomes the **Policy Manager** link).
2. Click the **Access System Configuration** tab.
3. Click the **AccessGate Configuration** link on the left panel.
4. Enter 'SOA_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 AccessGate for the domain you just created shows up (this will have the suffix _AG (for example, `SOA_EDG_AG`), click it, you will see the details of the AccessGate which you just created.

11.2.3.5 Updating the Host Identifier

The OAM Configuration Tool uses the value of the `app_domain` parameter to create a host identifier for the policy domain. This host identifier must be updated with all the host name variations for the host so that the configuration works correctly. Follow the steps below to update the host identifier created by the OAM Configuration Tool:

1. Navigate to the Access System Console by specifying the following URL in your web browser:

   ```
   http://hostname:port/access/oblix
   ```

   where *`hostname`* refers to the host where WebPass Oracle HTTP Server instance is running and *`port`* refers to the HTTP port of the Oracle HTTP Server instance.
2. When prompted for a username and password, log in as an administrator. Click **OK**.
3. On the Access System main page, click the **Access System Console** link.
4. On the Access System Console page, click the Access System Configuration tab.
5. On the Access System Configuration page, click **Host Identifiers** at the bottom left.
6. On the List all host identifiers page, click on the host identifier created by the OAM Configuration Tool. For example, select `SOA_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
- soahost1vhn1.mycompany.com:8010
- soahost2vhn1.mycompany.com:8010
- bamhost1.mycompany.com:9001
- bamhost2.mycompany.com:9001

- admin.mycompany.com:80
- adminvhn.mycompany.com:7001
- soahost1vhn1:8001
- soahost2vhn1:8001
- soahost1vhn1:8010
- soahost2vhn1:8010
- adminvhn:7001

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.

11.2.3.6 Updating the WebGate Profile

The OAM Configuration Tool populates the `Preferred_HTTP_Host` and hostname attributes for the WebGate profile that is created with the value of the `app_domain` parameter. Both these attributes must be updated with the proper values for the configuration to work correctly. Follow the steps below to update the WebGate profile created by the OAM CFG Tool.

1. Navigate to the Access System Console by specifying the following URL in your web browser:

   ```
   http://hostname:port/access/oblix
   ```

   where *`hostname`* refers to the host where WebPass Oracle HTTP Server instance is running and *`port`* refers to the HTTP port of the Oracle HTTP Server instance.

2. On the Access System main page, click the **Access System Console** link, then log in as an administrator.

3. On the Access System Console main page, click **Access System Configuration**, and then click the **Access Gate Configuration** link on the left pane to display the AccessGates Search page.

4. Enter the proper search criteria and click **Go** to display a list of AccessGates.

5. Select the AccessGate created by the OAM Configuration Tool. For example: **SOA_EDG_AG**.

6. On the AccessGate Details page, select **Modify** to display the Modify AccessGate page.
7. On the Modify AccessGate page, update:
   - **Hostname**: Update the hostname with the name of the computer where WebGate is running, for example: `webhost1.mycompany.com`.
   - **Preferred HTTP Host**: Update the Preferred_HTTP_Host with one of the hostname variations specified in the previous section, for example: `admin.mycompany.com: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.
9. Click **OK** to finish updating the configuration.
10. Verify the values displayed on the Details for AccessGate page to confirm that the updates were successful.

11.2.3.7 Adding Additional Access Servers

To assign an Access Server to the WebGate:

1. Log in as the Administrator on the Access System Console.
2. Navigate to the **Details** for AccessGate page, if necessary. From the Access System Console, select **Access System Configuration**, then **AccessGate Configuration**, then the link for the WebGate (**SOA_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.

11.2.3.8 Configuring Delegated Form Authentication

To configure the form authentication to redirect to the WebGate that was installed with the OAM installation, complete these steps:

1. Open the Access System Console.
2. In the Access System Configuration screen, select **Authentication Management** from the left-hand bar.
3. Select **OraDefaultFormAuthNScheme**.
4. Click **Modify**.
5. In the Challenge Redirect field, enter the host and port of the IDM installation; for example: `http://sso.mycompany.com`.

A WebGate should already be installed in the IDM installation. Refer to *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management* for details.

11.2.4 Installing and Configuring WebGate

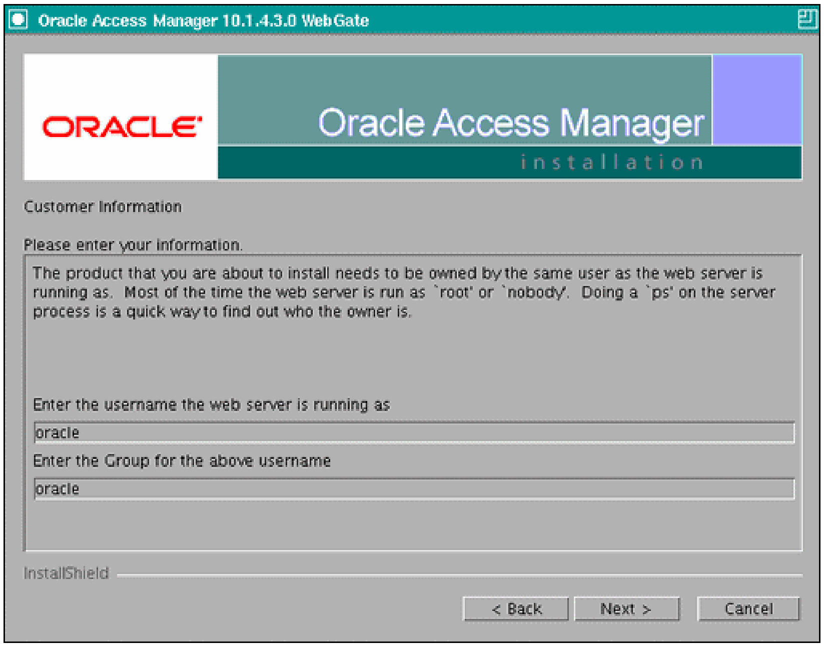
WebGate needs to be installed on each of the WEBHOST*n* machines in order to secure the web tier:

1. Launch the WebGate installer (see Section 1.5.5, "What to Install" for information on where to obtain it) using the following command:

   ```
   ./Oracle_Access_Manager10_1_4_3_0_linux_OHS11g_WebGate -gui
   ```

2. The Welcome screen is displayed. Click **Next**.
3. In the Customer Information screen (Figure 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

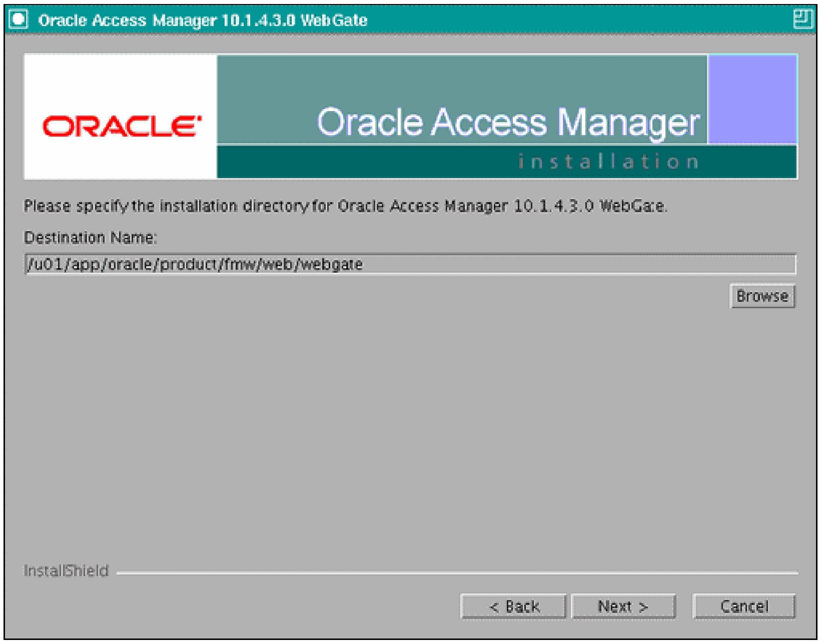
oracle

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



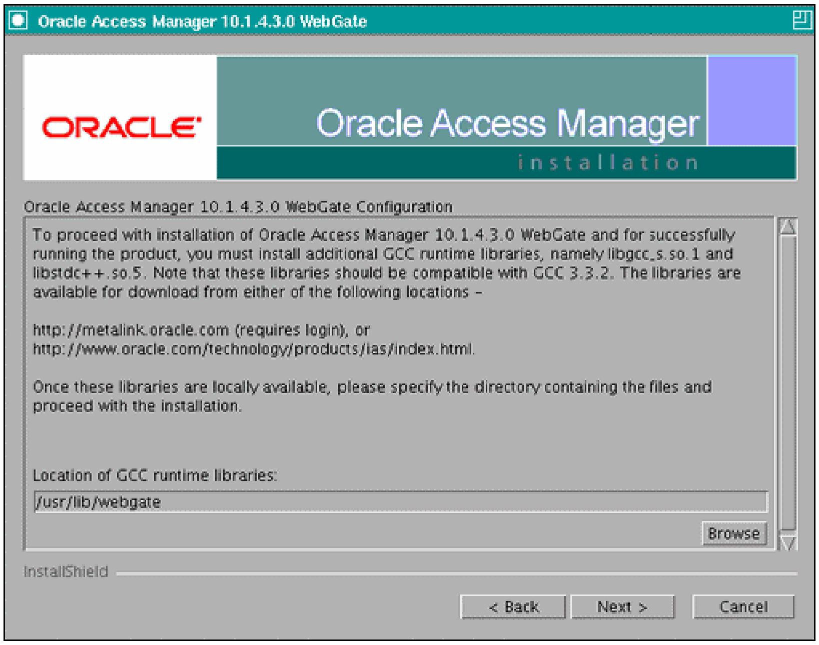
Please specify the installation directory for Oracle Access Manager 10.1.4.3.0 WebGate.

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.

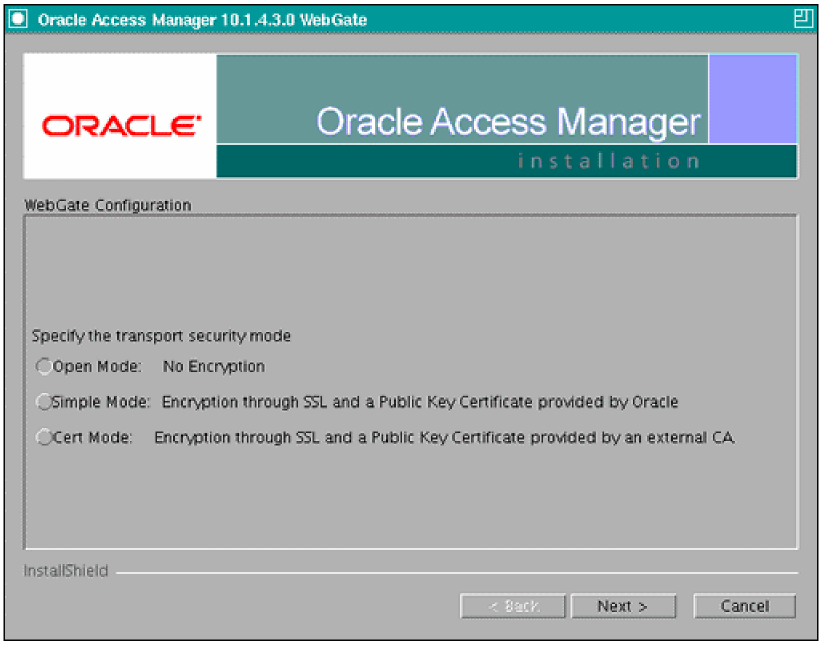
Once these libraries are locally available, please specify the directory containing the files and proceed with the installation.

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



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

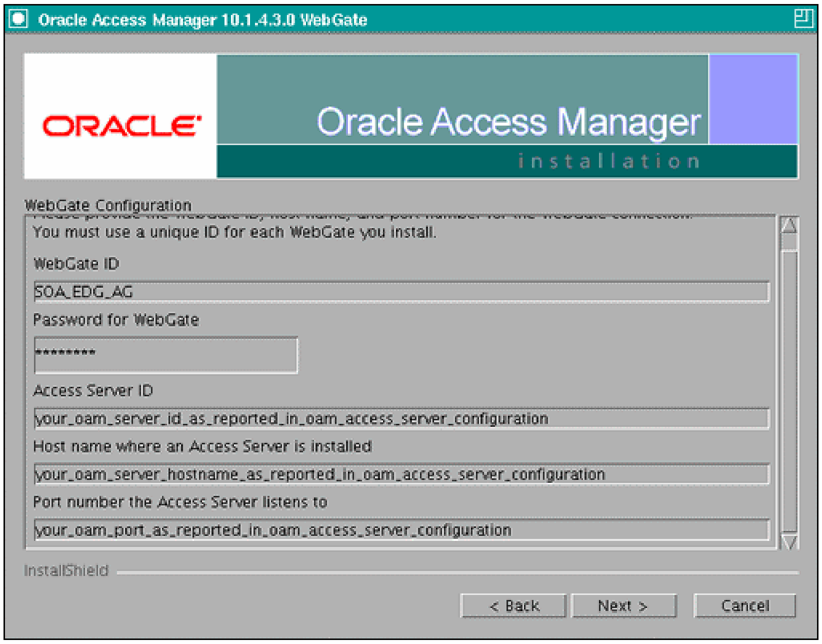
9. In the WebGate configuration screen, provide the details of the Access Server that will be used. You must provide the following information:

   - **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

SOA_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_ComponentName>
    ```

    For example:

    ```
    /u01/app/oracle/admin/ohs_instance2/config/OHS/ohs2/httpd.conf
    ```

    Click **Next** to continue.

12. In the next Configure Web Server page, a message informs you that the Web server configuration has been modified for WebGate. Click **Yes** to confirm.

13. Stop and start your Web server for the configuration updates to take effect. Click **Next** to continue.

14. In the next Configure Web Server screen, the following message is displayed: "If the web server is set up in SSL mode, then the httpd.conf file needs to be configured with the SSL related parameters. To manually tune your SSL configuration, please follow the instructions that come up". Click **Next** to continue.

15. In the next Configure Web Server screen, a message with the location of the document that has information on the rest of the product setup and Web server configuration is displayed. Choose **No** and click **Next** to continue.

16. The final Configure Web Server screen appears with a message to manually launch a browser and open the HTML document for further information on configuring your Web server. Click **Next** to continue.

17. The Oracle COREid Readme screen appears. Review the information on the screen and click **Next** to continue.

18. A message appears (along with the details of the installation) informing you that the installation was successful.

11.2.5 Changing the CacheControl Headers in the SOA_EDG_AG for Oracle BAM

Some Oracle BAM objects are required to be present in the browser´s cache or temp folder in order to be executed. When Oracle Access Manager is used as the Single Sign-On system for Oracle BAM, the HTTP cache header for Web pages is, by default, set to "no-cache" for security reasons. This prevents Internet Explorer from properly accessing some objects, as described in this Microsoft Knowledge Base note `http://support.microsoft.com/kb/316431`, and can cause exceptions while clicking different menu items in Oracle BAM's console. You must change the CacheControl headers settings in the SOA_EDG_AG Access Gate to prevent these errors. To change these settings:

1. Navigate to the Access System Console using the following URL:

   ```
   http://hostname:port/access/oblix
   ```

   Where *hostname* 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 on the left pane to display the AccessGates Search page.
4. Enter the proper search criteria and click **Go** to display a list of AccessGates.
5. Select the AccessGate created by the Oracle Access Manager configuration tool.
6. Click **Modify** at the bottom of the page.
7. In the **CachePragmaHeader** and **CacheControlHeader** fields, replace the **no-cache** field with **private**.
8. Click **Save** at the bottom of the page.

11.2.6 Configuring IP Validation for the Webgate

IP Validation determines if a client's IP address is the same as the IP address stored in the `ObSSOCookie` 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 balancer from the one front-ending the enterprise deployment. To configure your load balancer so that it is not validated in these cases, follow these steps:

1. Navigate to the Access System Console using the following URL:

   ```
   http://hostname:port/access/oblix
   ```

   Where the *hostname* 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, and 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 on the left pane to display the AccessGates Search page.
4. Enter the proper search criteria and click **Go** to display a list of AccessGates.
5. Select the AccessGate created by the Oracle Access Manager configuration tool.

6. Click **Modify** at the bottom of the page.
7. In the **IPValidationException** field, enter the address of the load balancer used to front-end the deployment.
8. Click **Save** at the bottom of the page.

11.2.7 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 includes the following topics:

- Section 11.2.7.1, "Back Up Configuration Files"
- Section 11.2.7.2, "Setting Up the OAM ID Asserter"
- Section 11.2.7.3, "Setting the Order of Providers"

11.2.7.1 Back Up Configuration Files

To be safe, first back up the relevant configuration files:

```
ORACLE_BASE/admin/<domain_name>/aserver/<domain_name>/config/config.xml
ORACLE_BASE/admin/<domain_name>/aserver/<domain_
name>/config/fmwconfig/jps-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.

11.2.7.2 Setting Up the OAM ID Asserter

To set up the OAM ID Asserter, complete these steps:

1. Log into Weblogic Console, if not already logged in.
2. Navigate to `SecurityRealms\<Default Realm Name>\Providers`.
3. Click **New** and Select "OAM Identity Asserter" from the dropdown menu.
4. Name the asserter (for example, "OAM ID Asserter") and click **Save**.
5. Click the newly added asserter to see the configuration screen for OAM Identity Asserter.
6. Set the control flag to 'REQUIRED' and click **Save**.
7. Open the **Provider Specific** tab to configure the following required settings:
   - **Primary Access Server:** provide OAM server endpoint information in HOST:PORT format.
   - **AccessGate Name:** name of the AccessGate (for example, `SOA_EDG_AG`).
   - **AccessGate Password:** password for the AccessGate (optional).
8. Save the settings.

11.2.7.3 Setting the Order of Providers

Reorder the OAM Identity Asserter, OID Authenticator, and Default Authenticator by ensuring that the control flag for each authenticator is set as follows:

- OAM Identity Asserter: REQUIRED

- OID LDAP Authenticator (or OVD LDAP Authenticator): SUFFICIENT
- Default Authenticator: SUFFICIENT

11.3 Oracle Access Manager 11g Integration

This section describes how to set up Oracle Access Manager 11g as the single sign-on solution for the Oracle SOA Enterprise Deployment topology.

This section 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, "Install WebGate,"
- Section 11.3.4, "Register the WebGate Agent,"
- Section 11.3.5, "Set Role Members for BPMWorkflowAdmin Application Role in soa-infra,"
- Section 11.3.6, "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 SOA installation with an existing OAM 11g installation and the underlying directory service. Oracle recommends using either Oracle Internet Directory (OID), Oracle Virtual Directory (OVD), or both of these directory services.

> **Note:** The SOA topology described in this guide uses a Single Sign-On configuration where both the SOA System and the Single Sign-On System are in the same network domain (mycompany.com). For a multi-domain configuration, please refer to the required configuration steps in "Chapter 7, Configuring Single Sign-On," of the *Oracle Access Manager Access Administration Guide*.

11.3.2 Prerequisites for Oracle Access Manager

The setup for Oracle Access Manager (OAM) assumes an existing OAM installation complete with Access Managers and a policy protecting the Policy Manager. For more information on installing and configuring an OAM installation, see *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*. This setup includes a directory service such as Oracle Internet Directory (OID) either as a stand-alone or as part of an Oracle Virtual Directory (OVD) configuration. This chapter provides the necessary steps for configuring your SOA installation with either OID or OVD.

In addition, the OAM installation should have its own Web server configured with a WebGate. This section also provides the steps for using the OAM Web server as a delegated authentication server.

11.3.3 Install WebGate

You must install a WebGate on each of the WEBHOST machines where an HTTP Server has already been installed. Section 11.3.3and Section 11.3.4 should be repeated for each WEBHOST in the deployment environment.

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 Web site:

http://gcc.gnu.org/

For Linux 32-bit 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 GCC third-party 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 11g Webgate for Oracle Access Manager is included in the webgate.zip file.

To start the installation wizard:

1. Extract the contents of the webgate.zip file to a directory. By default, this directory is named `webgate`.
2. Move to the Disk1 directory under the webgate folder.
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 appears.

Installation Flow and Procedure

If you need additional help with any of the installation screens, click **Help** to access the online help.

To install Oracle HTTP Server 11g 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, 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 11g Webgate for Oracle Access Manager:

1. Move to the following directory under your Oracle Home for Webgate:

   ```
   $ cd Webgate_Home/webgate/ohs/tools/deployWebGate
   ```

2. On the command line, run the following command 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
   ```

   Where *Webgate_Oracle_Home* is the directory where you have installed Oracle HTTP Server Webgate and created as the Oracle Home for Webgate, as in the following 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, as in the following 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 to copy the `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, as in the following 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, as in the following 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, as in the following example:

   ```
    Edithttpconf.log
   ```

11.3.4 Register the WebGate Agent

This section describes the procedures for registering the WebGate Agent.

11.3.4.1 The RREG Tool

The RREG tool is part of the OAM 11g 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 -xvf RREG.tar
   ```

You can find the tool that is used to register the agent in the following location:

```
RREG_Home/bin/oamreg.sh
```

RREG_Home is the directory to which you extracted the contents of RREG.tar.gz/rreg.

The RREG Configuration Tool provides a way to register protected and public resources into the OAM system. The list of protected resources to be added to the OAM system is as follows:

```
/integration/worklistapp
/integration/worklistapp/.../*
/workflow/sdpmessagingsca-ui-worklist
/workflow/sdpmessagingsca-ui-worklist/.../*
/b2bconsole
/b2bconsole/.../*
/sdpmessaging/userprefs-ui
/sdpmessaging/userprefs-ui/…/*
/DefaultToDoTaskFlow
/DefaultToDoTaskFlow/…/*
/em
/em/.../*
/console
/console/.../*
/soa/composer
/soa/composer/.../
/OracleBAM (For BAM systems only)
/OracleBAM/…/* (For BAM systems only)
/bpm/composer (For BPM systems only)
/bpm/composer/…/* (For BPM systems only)
/bpm/workspace
/bpm/workspace/…/*
/soa-infra
/soa-infra/deployer
/soa-infra/events/edn-db-log
/soa-infra/cluster/info
/inspection.wsil/
```

The list of public resources is:

```
/soa-infra/services/.../*
/soa-infra/directWSDL
/soa-infra/directWSDL/.../*
/OracleBAMWS
/OracleBAMWS/.../*
/ucs/messaging/webservice
/ucs/messaging/webservice/.../*
```

Where "/…/*" implies all resources under the base url context.

11.3.4.2 Updating the OAM11gRequest file

In the *RREG_Home*/input directory there are template files named `OAM11gRequest.xml`. This file should be copied and edited in order to create the policies for the SOA installation. After editing, the file should look as follows:

> **Note:** Replace $$webtierhost$$, $$oamadminserverport$$, and $$oamhost$$ with the hostnames 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://$$oamhost$$:$$oamadminserverport$$</serverAddress>
    <hostIdentifier>$$webtierhost$$_soa</hostIdentifier>
    <agentName>$$webtierhost$$_soa</agentName>
    <applicationDomain>$$webtierhost$$_soa</applicationDomain>
    <cachePragmaHeader>private</cachePragmaHeader
    <cacheControlHeader>private</cacheControlHeader>
    <ipValidation>1</ipValidation>
    <logOutUrls>
        <url></url>
    </logOutUrls>
    <protectedResourcesList>
        <resource>/integration/worklistapp</resource>
        <resource>/integration/worklistapp/…/*</resource>
        <resource>/workflow/sdpmessagingsca-ui-worklist</resource>
        <resource>/workflow/sdpmessagingsca-ui-worklist/…/*</resource>
        <resource>/b2bconsole</resource>
        <resource>/b2bconsole/…/*</resource>
        <resource>/sdpmessaging/userprefs-ui</resource>
        <resource>/sdpmessaging/userprefs-ui/…/*</resource>
        <resource>/DefaultToDoTaskFlow</resource>
        <resource>//DefaultToDoTaskFlow/…/*</resource>
        <resource>/em</resource>
        <resource>/em/…/*</resource>
        <resource>/console</resource>
        <resource>/console/…/*</resource>
        <resource>/soa/composer</resource>
        <resource>/soa/composer/…/*</resource>
        <resource>/OracleBAM </resource><!-- (For BAM systems only) -->
        <resource>/OracleBAM/…/*</resource><!-- (For BAM systems only) -->
        <resource>/bpm/composer</resource> <!-- (For BPM systems only) -->
        <resource>/bpm/composer/…/*</resource> <!-- (For BPM systems only) -->
        <resource>/bpm/workspace</resource><!-- (For BPM systems only) -->
        <resource>/bpm/workspace/…/*</resource><!-- (For BPM systems only) -->
        <resource>/soa-infra</resource>
        <resource>/soa-infra/deployer</resource>
        <resource>/soa-infra/deployer/…/*</resource>
        <resource>/soa-infra/events/edn-db-log</resource>
        <resource>/soa-infra/events/edn-db-log/…/*</resource>
        <resource>/soa-infra/cluster/info</resource>
        <resource>/soa-infra/cluster/info/…/*</resource>
        <resource>/inspection.wsil</resource>
    </protectedResourcesList>
    <publicResourcesList>
        <resource>/soa-infra/services/.../*</resource>
        <resource>/soa-infra/directWSDL</resource>
        <resource>/OracleBAMWS</resource> <!-- (For BAM systems only) -->
        <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/SOAOAM11GRequest.xml
```

The run should look as follows:

```
------------------------------------------------
Welcome to OAM Remote Registration Tool!
Parameters passed to the registration tool are:
Mode: inband
Filename: MW_HOME/Oracle_IDM1/oam/server/rreg/input/SOAOAM11gRequest.xml
Enter your agent username:weblogic
Username: weblogic
Enter agent password:
Do you want to enter a Webgate password?(y/n):
y
Enter webgate password:
Enter webgate password again:
Password accepted. Proceeding to register..
Aug 16, 2010 1:22:30 AM
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:WEBHOST1_soa
URL String:WEBHOST1_soa
Registering in Mode:inband
Your registration request is being been 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 Copy Access files to WEBHOSTs

The following two files are generated in *RREG_Home*/output/$$webtierhost$$_soa:

- ObAccessClient.xml
- cwallet.sso

Copy these files to the webgate instance (*Webgate_Instance_Home*/config/OHS/ohsN/webgate/config/) location on the WEBHOST machine.

11.3.5 Set Role Members for BPMWorkflowAdmin Application Role in soa-infra

When associating the domain with a identity store that does not contain the user "weblogic", you must assign some other valid user into the application role BPMWorkflowAdmin. To assign the role to a valid user:

1. Create a user in LDAP Store, in this case named **SOAAdmin**. This user will be assigned the role.
2. Assign the role. This can be done using wlst from the SOA Oracle home:

   For example:

   ```
   cd ORACLE_HOME/common/bin/
   wlst.sh

   connect('weblogic','weblogic', 'SOAADMINHOST:7001')
   revokeAppRole(appStripe="soa-infra", appRoleName="BPMWorkflowAdmin",
   principalClass="oracle.security.jps.service.policystore.ApplicationRole",
   principalName="SOAAdmin")
   grantAppRole(appStripe="soa-infra", appRoleName="BPMWorkflowAdmin",
   principalClass="weblogic.security.principal.WLSUserImpl",
   principalName="SOAAdmin")
   ```

11.3.6 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 includes the following topics:

- Section 11.3.6.1, "Back Up Configuration Files"
- Section 11.3.6.2, "Setting Up the OAM ID Asserter"
- Section 11.3.6.3, "Setting the Order of Providers"

11.3.6.1 Back Up Configuration Files

To be safe, first back up the relevant configuration files:

```
ORACLE_BASE/admin/domain_name/aserver/domain_name/config/config.xml
ORACLE_BASE/admin/domain_name/aserver/domain_name/config/fmwconfig/jps-con
fig.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.

11.3.6.2 Setting Up the OAM ID Asserter

To set up the OAM ID Asserter:

1. Log into Weblogic Console, if not already logged in.
2. Click **Lock and Edit.**
3. Navigate to **SecurityRealms**, **<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 **Save**.
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 as admin to WLST console and run the following command:

```
addOAMSSOProvider(loginuri="/${app.context}/adfAuthentication",logouturi="oams
so/logout.html")
```

11.3.6.3 Setting the Order of Providers

Reorder the OAM Identity Asserter, OID Authenticator, and Default Authenticator by ensuring that the control flag for each authenticator is set as follows:

- OAM Identity Asserter: REQUIRED
- OID LDAP Authenticator (or OVD LDAP Authenticator): SUFFICIENT
- Default Authenticator: SUFFICIENT

11.4 Backing Up the Installation

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

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

1. Back up the web tier:

   a. Shut down the instance using `opmnctl`.

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

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

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

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

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

   d. Start the instance using `opmnctl`:

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

2. Back up the AdminServer domain directory. Perform a backup to save your domain configuration. The configuration files all exist under the `ORACLE_BASE/admin/<domain_name>` directory.

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

12 Miscellaneous

This chapter contains the following sections:

12.1 Recovering Failed BPEL and Mediator Instances

This section describes how to check and recover failed instances in BPEL, Mediator and other service engines.

> **Note:** For the steps that require you to run SQL statements, you connect to the database as the soainfra schema.

- To check for recoverable instances, run the following SQL statements in the database:

```
// Find recoverable activities
SQL> select * from work_item where state = 1 and execution_type != 1;

// Find recoverable invoke messages
SQL> select * from dlv_message where dlv_type = 1 and state = 0;

// Find recoverable callback messages
SQL> select * from dlv_message where dlv_type = 2 and (state = 0 or state = 1);
```

- To recover failed BPEL instances:

  In Enterprise Manager, select **Farm_<domain_name>**, then expand **SOA**, then right click on **soa-infra (server_soa)**, then **Service Engine**, then **BPEL**, and then **Recovery**.

- To recover a failed Mediator composite:

In Enterprise Manager, select **Farm_<domain_name>**, then expand **SOA**, then right-click on **soa-infra (server_soa)**, then **Service Engine**, then select **Mediator**, and then **Fault**.

- To check for rejected messages:

  ```
  SQL> select * from rejected_message
  ```

- To check data in the instance tracking table, run the following SQL query:

  ```
  SQL> select ID, STATE from COMPOSITE_INSTANCE where CREATED_TIME > datetime
  ```

  where *datetime* specifies the date and time to narrow the query. For example:

  ```
  '04-NOV-09 03.20.52.902000000 PM'
  ```

  The adapter enters data into the `COMPOSITE_INSTANCE` table before anywhere else.

  When the adapter publishes data to the Adapter BC, the BC inserts an entry into the `COMPOSITE_INSTANCE` table with STATE as `0`. After the message has been processed, the STATE becomes `1`. In case of errors, STATE >= `2`.

12.2 Configuring Web Services to Prevent Denial of Service and Recursive Node Attacks

Configure `SCABindingProperties.xml` and `oracle-webservices.xml` to configure Web services against denial of service attack and recursive node attack.

Configuring SCABindingProperties.xml

To prevent denial of service attacks and recursive node attacks, set the envelope size and nesting limits in `SCABBindingProperties.xml` as illustrated in Example 12–1.

Example 12–1 Configuring Envelope Size and Nesting Limits in SCABBindingProperties.xml

```
<bindingType type="ws">
        <serviceBinding>
                <bindingProperty>
                     <name>request-envelope-max-kilobytes</name>
                    <type>xs:integer</type>
                     <defaultValue>-1</defaultValue>
                 </bindingProperty>
                 <bindingProperty>
                   <name>request-envelope-nest-level</name>
                     <type>xs:integer</type>
                    <defaultValue>-1</defaultValue>
                 </bindingProperty>
        </serviceBinding>
```

Configuring oracle-webservices.xml

For standalone Web services, configure the envelope size and nesting limits in `oracle-webservices.xml`. For example:

```
<request-envelope-limits kilobytes="4" nest-level="6" />
```

> **Note:** Setting the envelope and nesting limits to extremely high values, or setting no values at all, can lead to denial of service attacks.

12.3 Running the SOA Fusion Order Demo Application in an Enterprise Deployment Environment

To run the Fusion Order Demo Application (FOD) in an environment described in the Enterprise Deployment Guide, you must update the FOD's `createJMSResources.seed` file (located at `./bin/templates`) by replacing the *SOAJMSModule* with *SOAJMSModuleUDDS* as the system resource name before running the ANT task that creates the JMS artifacts required for the FulFillment Mediator demo application.

12.4 Oracle Business Activity Monitoring (BAM) Configuration Properties

To increase or decrease the number of times BAM retries the in-flight transactions after an Oracle RAC failover, change the `MaxDBNodeFailoverRetries` setting from its default of 5 times to another value. However, it is a best practice to maintain the default settings for `UseDBFailover` and `MaxDBNodeFailoverRetries`. To disable BAM's Oracle RAC failover retry support, set `UseDBFailover` to false. (The default value for this setting is true.) For information on using these settings, see "Oracle BAM Configuration Property Reference" in *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite*.

12.5 Using Shared Storage for Deployment Plans and SOA Infrastructure Applications Updates

When redeploying a SOA infrastructure application or resource adapter within the SOA cluster, the deployment plan along with the application bits should be accessible to all servers in the cluster. SOA applications and resource adapters are installed using nostage deployment mode. Because the administration sever does not copy the archive files from their source location when the nostage deployment mode is selected, each server must be able to access the same deployment plan. Use `ORACLE_BASE/admin/<domain_name>/<cluster_name>/dp` as the location for the deployment plan and applications, which should be accessible from all nodes in the Enterprise Deployment topology (as per the recommendation in Section 2.3, "Shared Storage and Recommended Directory Structure").

Index