Oracle® Fusion Middleware

Enterprise Deployment Guide for Oracle SOA Suite

11*g* Release 1 (11.1.1.7.0)

E12036-13

August 2014

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

ORACLE®

Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite, 11*g* Release 1 (11.1.1.7.0)

E12036-13

Contents

3 Preparing the Network for an Enterprise Deployment

4 Preparing the File System for an Enterprise Deployment

5 Preparing the Database for an Enterprise Deployment

6 Installing the Software for an Enterprise Deployment

7 Configuring the Web Tier for an Enterprise Deployment

8 Creating a Domain for an Enterprise Deployment

9 Extending the Domain for SOA Components

10 Extending the Domain to Include Oracle BPM

11   Extending a SOA Domain to Oracle Service Bus

14   Configuring Server Migration for an Enterprise Deployment

15   Integrating an Enterprise Deployment with Oracle Identity Management

16   Managing the Topology for an Enterprise Deployment

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

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

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/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Related Documents

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 About the Enterprise Deployment Guide

The Enterprise Deployment Guide is an Oracle best practices blueprint based on proven Oracle high-availability and security technologies and recommendations for an Oracle SOA enterprise deployment. 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, see the Oracle Database High Availability page on Oracle Technology Network at http://www.oracle.com/technetwork/database/features/availability/maa-best-practices-155366.html.

> **Note:** The Enterprise Deployment Guide for Oracle SOA focuses on enterprise deployments in Linux environments. However, you can also implement enterprise deployments using UNIX and Windows environments.

1.2 Enterprise Deployment Terminology

This section identifies enterprise deployment terminology used in the guide.

- **Oracle home**: An Oracle home contains installed files necessary to host a specific product. For example, the SOA Oracle home contains a directory that contains binary and library files for Oracle SOA Suite. An Oracle home resides within the directory structure of the Middleware home. Each Oracle home can be associated with multiple Oracle instances or Oracle WebLogic Server domains.

- **Oracle Common home**: 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 Oracle, HP, IBM, and Dell. While the number of nodes that can be configured in a hardware cluster is vendor dependent, for the purpose of Oracle Fusion Middleware high availability, only two nodes are required. Hence, this document assumes a two-node hardware cluster for high availability solutions employing a hardware cluster.

- **cluster agent**: The software that runs on a node member of a hardware cluster that coordinates availability and performance operations with other nodes. Clusterware provides resource grouping, monitoring, and the ability to move services. A cluster agent can automate the service failover.

- **clusterware**: 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 are 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.

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

This section includes the following topics:

- Section 1.3.1, "Built-in Security"

- Section 1.3.2, "High Availability"

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.

- 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.

2

Introduction to the Enterprise Deployment Reference Topology

This chapter describes and illustrates the enterprise deployment reference topologies described in this guide. The road map for installation and configuration directs you to the appropriate chapters for the tasks you need to perform. Use this chapter to help you plan your Oracle SOA enterprise deployment.

This chapter includes the following topics:

2.1 Overview of Enterprise Deployment Reference Topologies

This diagrams in this section illustrate the three possible enterprise deployment topologies described in this guide. Use this section to plan your enterprise deployment topology.

This section covers these topics:

2.1.1 Reference Topologies Documented in the Guide

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

Figure 2–3 or with Oracle Service Bus, as shown in Figure 2–4.

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

2.1.1.1 MySOACompany Topology with Oracle Access Manager

Figure 2–1 illustrates a MySOACompnay topology that includes Oracle Access Manager.

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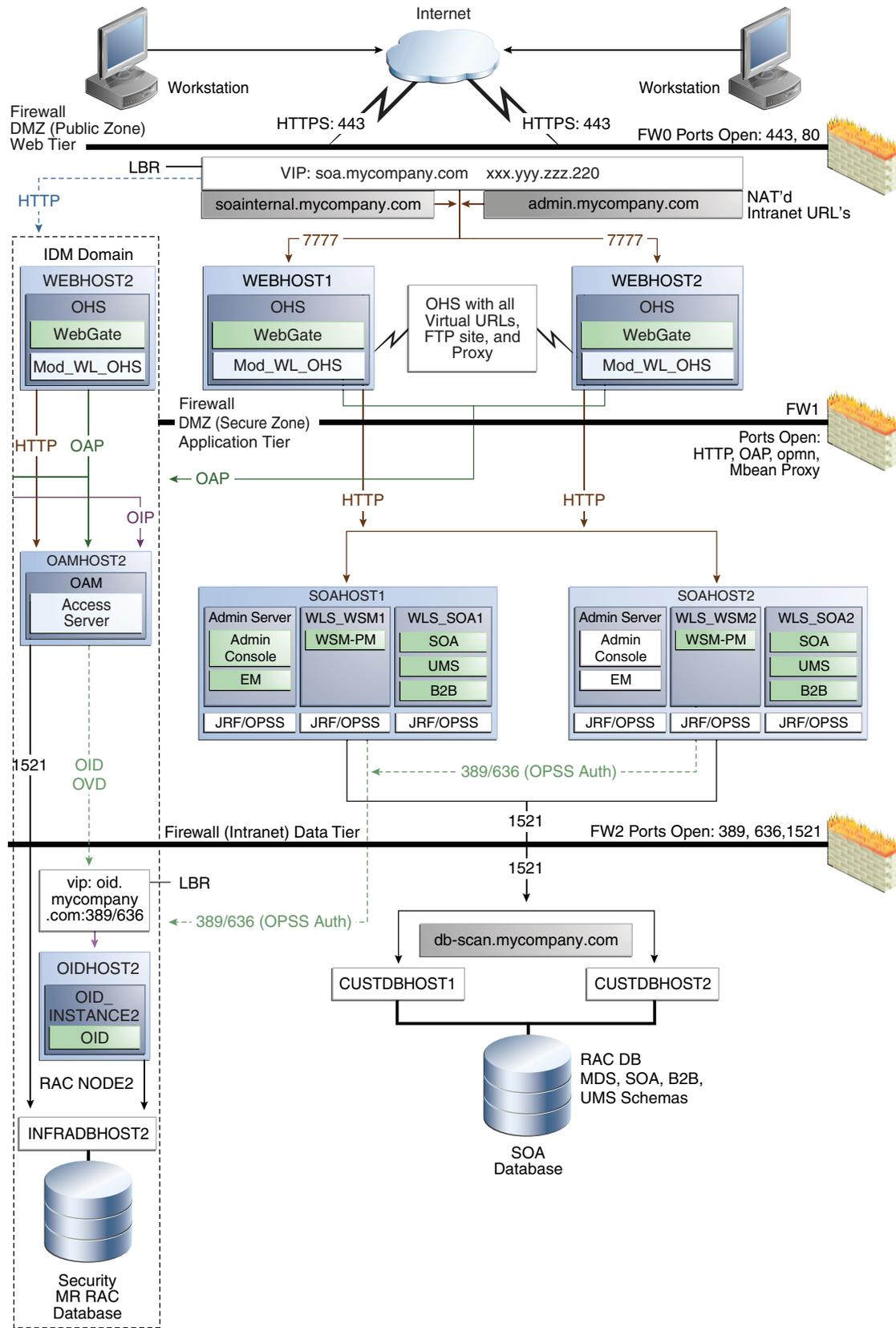
2.1.1.2 MySOACompany Topology with Oracle Access Manager and Business Activity Monitoring

Figure 2–2 illustrates a MySOACompany topology that includes Oracle Access Manager and Business Activity Monitoring.

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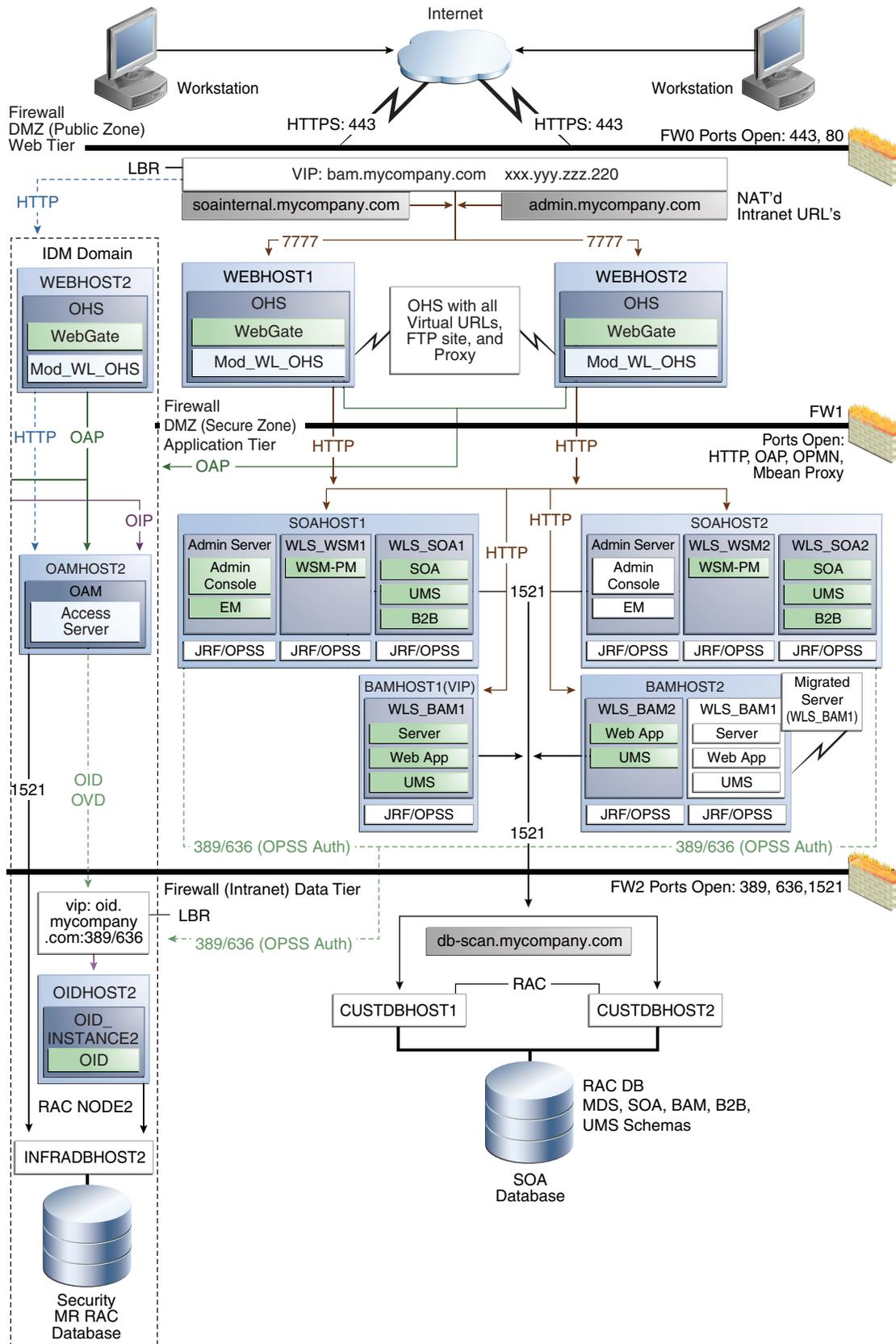
2.1.1.3 MySOACompany Topology with Oracle Access Manager and Oracle BPM

Figure 2–3 illustrates a MySOACompany Topology that includes Oracle Access Manager and Oracle BPM.

Figure 2–3 illustrates a MySOACompany Topology that includes Oracle Access Manager and Oracle BPM.

Figure 2–3 MySOACompany Topology with Oracle Access Manager and BAM
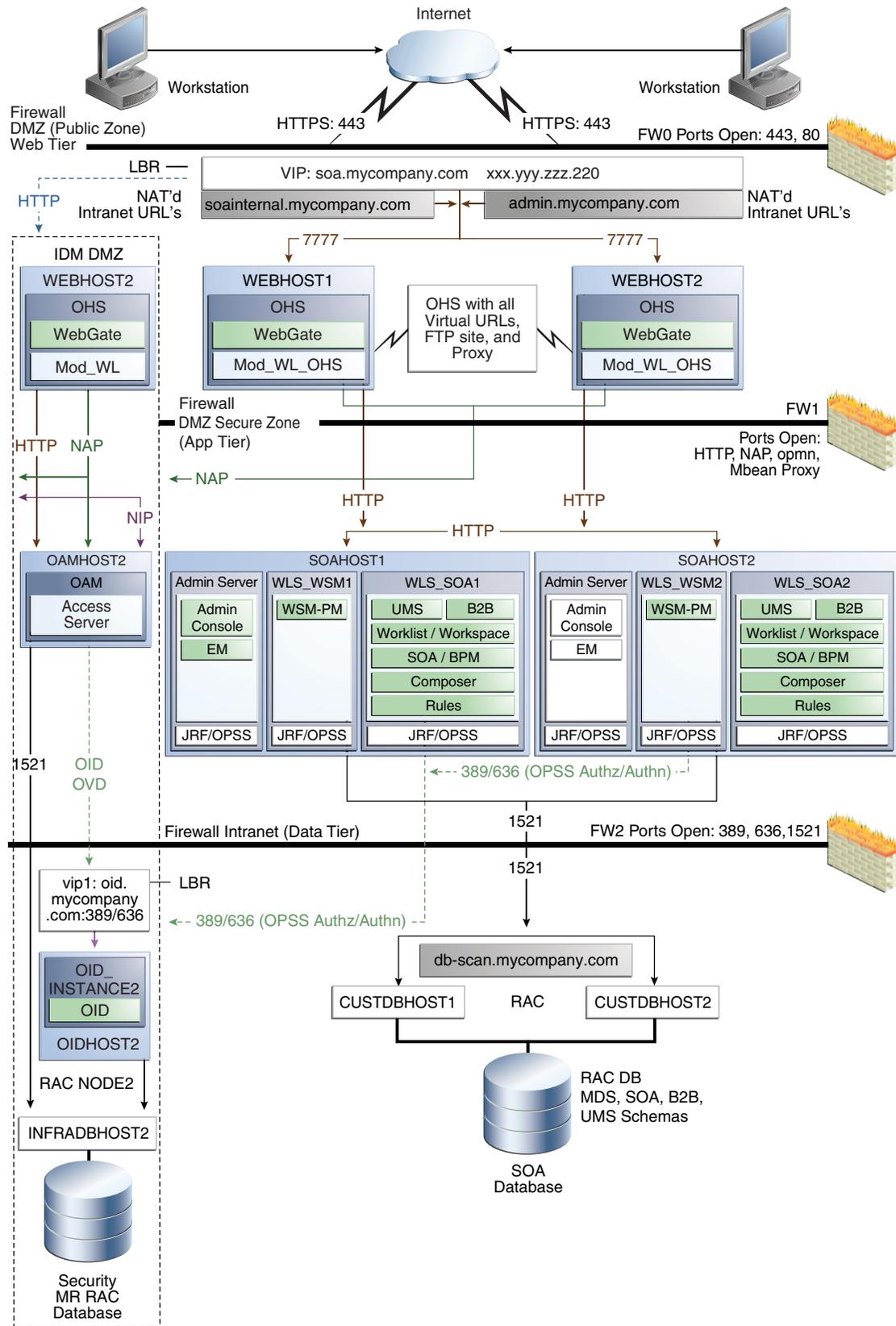
2.1.1.4 MySOACompany Topology with Oracle Service Bus

Figure 2–4 illustrates a MySOACompany Topology that includes Oracle Service Bus.

Figure 2–4   MySOACompany Topology with Oracle Service Bus
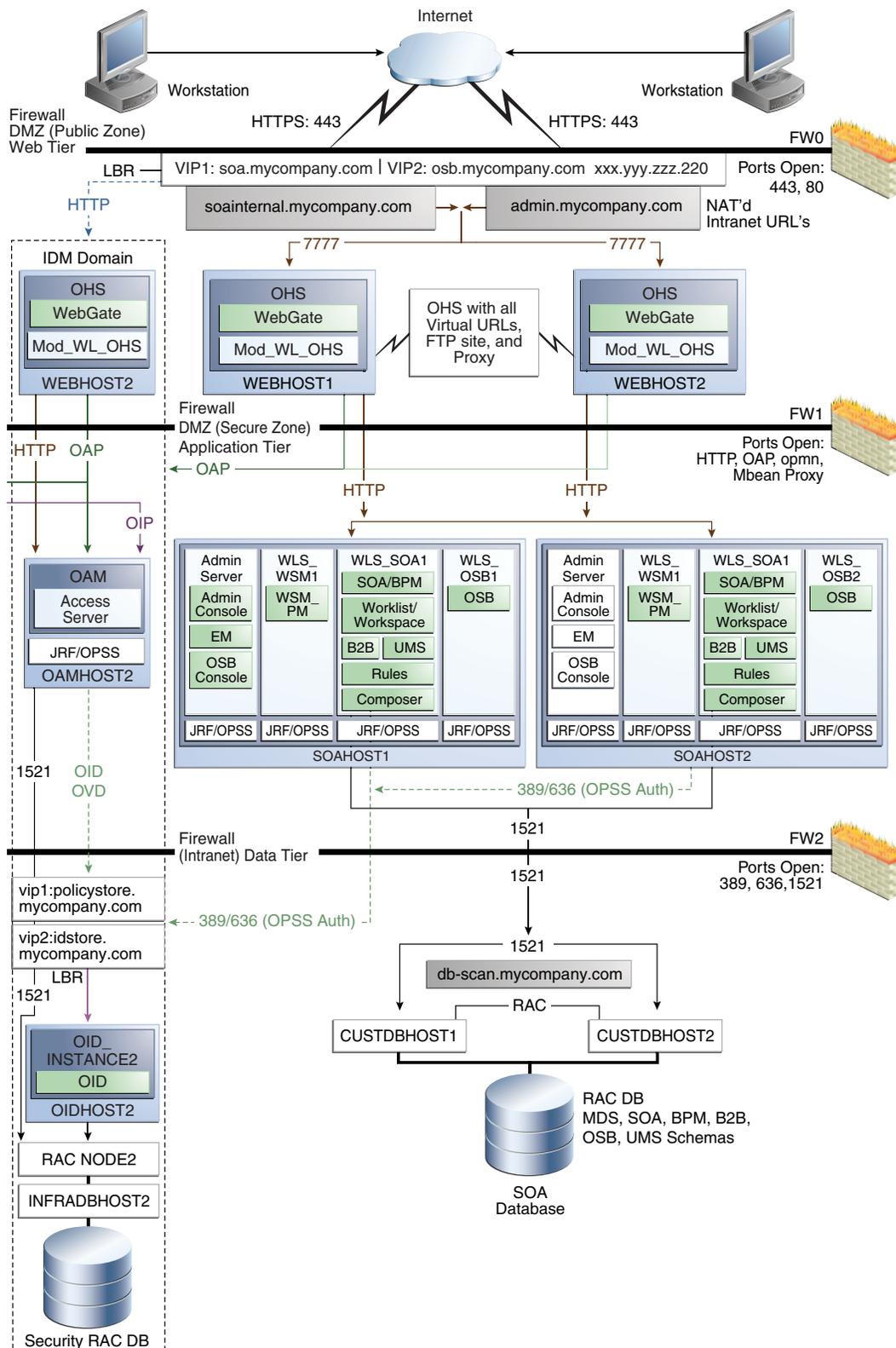
2.1.2 About Oracle Identity Management Integration

Integration with the Oracle Identity Management system is an important aspect of the enterprise deployment architecture. This integration provides features such as single sign-on, integration with Oracle Platform Security Services, centralized identity and credential store, and authentication for the WebLogic domain. 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.

2.1.3 About the Web Tier Nodes

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

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

The following is a list of benefits provided by using Oracle HTTP Server as an intermediate point between the load balancer and the different WebLogic Servers:

- It provides a sacrificial area/DMZ. This is a common requirement in security audits and is a major problem with load balancer/WebLogic systems. If a load balancer routes directly to the WebLogic Server, request move from the load balancer to the application tier in one single HTTP jump, causing security concerns.

- It allows the WebLogic Server cluster membership to be reconfigured (new servers added, others removed) without having to change the Web server configuration (as long as at least some of the servers in the configured list remain alive). The plug-in learns about the cluster membership and directs work accordingly.

- Faster fail-over in the event of WebLogic Server instance failure. The plug-in actively learns about the failed WebLogic Server instance using information supplied by its peers, It avoids the failed server until the peers notify the plug-in that it is again available. Load balancers are typically more limited.

- Oracle HTTP Server delivers static content more efficiently and faster than WebLogic Server.

- HTTP redirection over and above what WebLogic Server provides. You can use Oracle HTTP Server as a front end against many different WebLogic Server clusters, and perhaps do content based routing.

- If SSO is required, only Oracle HTTP Server (not WebLogic Server) supports Oracle Identity Management.

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.

2.1.4 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 termination and it is required for this Enterprise Deployment.

- SSL acceleration, which refers to off loading the public-key encryption algorithms involved in SSL transactions to a hardware accelerator. This feature is recommended, but not required.

2.1.5 About the Application Tier

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

In this tier, two nodes SOAHOST1 and SOAHOST2 run Oracle WebLogic Server configured with managed servers for running 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. The active-passive configuration of the Administration Server is necessary because only one Administration Server can be running within a domain. In the illustration, the Administration Server on SOAHOST1 is currently in the active state, but you can failover manually to the Administration Server on SOAHOST2.

For information, see Section 8.8, "Verifying Manual Failover of the Administration Server." 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.)

2.1.6 About the 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, BAM, and Oracle Service Bus components. The SOA, BAM and Oracle Service Bus 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.

2.1.7 About the Unicast Requirement for Communication

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.

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

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

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

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

- All members of a WebLogic cluster must use the same message type. Mixing between multicast and unicast messaging is not allowed.

- Individual cluster members cannot override the cluster messaging type.

- The entire cluster must be shut down and restarted to change the message modes (from unicast to multicast or from multicast to unicast).

- JMS topics configured for multicasting can access WebLogic clusters configured for unicast because a JMS topic publishes messages on its own multicast address that is independent of the cluster address. However, the following considerations apply:

  – The router hardware configurations that allow unicast clusters may not allow JMS multicast subscribers to work.

  – JMS multicast subscribers need to be in a network hardware configuration that allows multicast accessibility. (That is, JMS subscribers must be in a multicast-enabled network to access multicast topics.)

2.2 Hardware Requirements for an Enterprise Deployment on Linux

Read the Oracle Fusion Middleware System Requirements and Specifications document to ensure that your environment meets the minimum installation requirements for the products you are installing. This document contains information related to hardware and software requirements, minimum disk space and memory requirements, database schema requirements, and required system libraries, packages, or patches.

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

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.

Table 2–1   Typical Hardware Requirements

Server	Disk	Memory	TMP Directory	Swap
Database	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*	10 GB	4 GB	Default	Default

Table 2–1   (Cont.) Typical Hardware Requirements

Server	Disk	Memory	TMP Directory	Swap
SOAHOST*n* (SOA only)	10 GB[1]	4 GB	Default	Default
SOAHOST*n* (SOA and OSB)	11 GB[2]	6 GB	Default	Default
BAMHOST*n*	10 GB[3]	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] For a shared storage Middleware home configuration, two installations suffice by making a total of 20 GB independently of the number of slots.

[3] BAM can reuse Middleware home binaries from the SOA installation in shared storage.

2.3 Identifying the Software Components to Install

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

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

Table 2–2   Components and Installation Sources

Component	Details
Oracle Database 10*g* or 11*g*	Oracle Database 10*g* distribution (10.2.0.4 or later SE or EE version of the database) using the AL32UTF8 character set.
	Oracle Database Server 11*g* distribution (11.1.0.7 or later SE or EE version of the database), using the AL32UTF8 character set.
Repository Creation Utility (RCU) (The RCU is generic and not platform specific on different Linux variations. The same bits are installed for Linux on 32 and 64 bit installations.)	Oracle Fusion Middleware Repository Creation Utility 11*g* (11.1.1.7) distribution
Oracle WebLogic Server (WLS)	Oracle WebLogic Server (10.3.6) distribution
Oracle HTTP Server (OHS)	Oracle Fusion Middleware WebTier and Utilities 11*g* (11.1.1.7) distribution
Oracle SOA Suite	Oracle SOA Suite 11*g* (11.1.1.7) distribution
Oracle Access Manager (OAM) WebGate	WebGate 10*g* (10.1.4.3) for OAM 10*g* or WebGate 11*g* (11.1.1.2 or later) for OAM 11*g*.
Oracle Virtual Directory (OVD)	Oracle Identity and Access Management 11*g* (11.1.1.5 or later) distribution
Oracle Internet Directory (OID)	Oracle Identity and Access Management 11*g* (11.1.1.5 or later) distribution
Oracle Service Bus (OSB)	Oracle Service Bus 11*g* (11.1.1.7) distribution

2.4 About 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 servers as described in Section 4.3, "About Recommended Locations for the Different

Directories." Derived from this, and for integrity and consistency purposes, Oracle requires the use of an LDAP as policy and credential store in the 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 15.3, "Configuring the Policy Store."

2.5 Clock Synchronization

The clocks of all servers participating in the cluster must be synchronized to within one second difference to enable proper functioning of jobs, adapters, and Oracle B2B. To accomplish this, use a single network time server and then point each server to that network time server.

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

2.6 Road Map for the Reference Topology Installation and Configuration

Before beginning your Oracle SOA enterprise deployment, review the flow chart in Figure 2–5. This flow chart illustrates the high-level process for completing the enterprise deployment documented in this guide. Table 2–3 describes the steps in the flow chart and directs you to the appropriate section or chapter for each step.

This section covers the following topics:

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

2.6.1 Flow Chart of the Oracle SOA Enterprise Deployment Process

Figure 2–5 provides a flow chart of the Oracle SOA enterprise deployment process. Review this chart to become familiar with the steps that you must follow, based on the existing environment.

Figure 2–5   Flow Chart of the Oracle SOA Enterprise Deployment Process
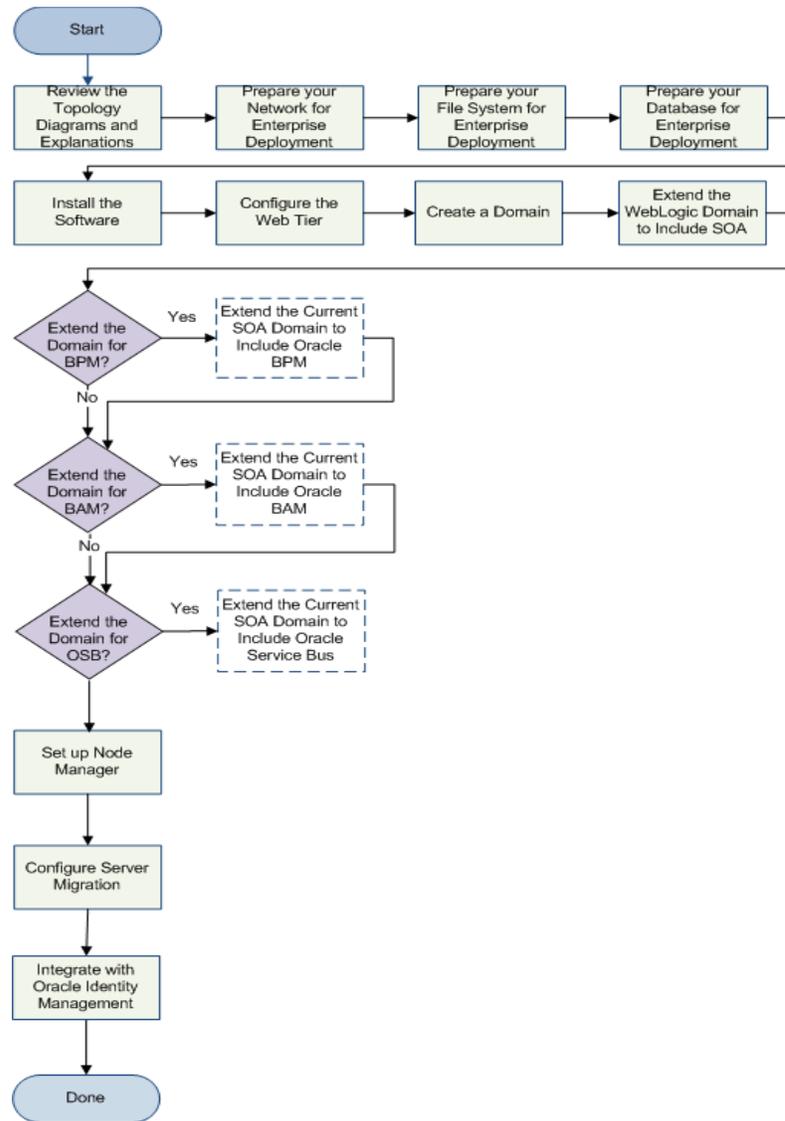
2.6.2 Steps in the Oracle SOA Enterprise Deployment Process

Table 2–3 describes each of the steps in the enterprise deployment process flow chart for Oracle SOA, shown in Figure 2–5. The table also provides information on where to obtain more information on each step in the process.

Table 2–3   Steps in the Oracle SOA Enterprise Deployment Process

Step	Description	More Information
Prepare your Network for Enterprise Deployment	To prepare your network for an enterprise deployment, understand concepts, such as virtual server names and IPs and virtual IPS, and configure your load balancer by defining virtual host names.	Chapter 3, "Preparing the Network for an Enterprise Deployment"
Prepare your File System for Enterprise Deployment	To prepare your file system for an enterprise deployment, review the terminology for directories and directory environment variables, and configure shared storage.	Chapter 4, "Preparing the File System for an Enterprise Deployment"
Prepare your Database for Enterprise Deployment	To prepare your database for an enterprise deployment, review database requirements, create database services, load the metadata repository, in the Oracle RAC database, configure SOA schemas for transactional recovery privileges, and back up the database.	Chapter 5, "Preparing the Database for an Enterprise Deployment"
Install the Software	Install Oracle HTTP Server, Oracle WebLogic Server, Oracle Fusion Middleware, and apply patchsets to Oracle Fusion Middleware components.	Chapter 6, "Installing the Software for an Enterprise Deployment"
Configure the Web Tier	Configuring Oracle HTTP Server with the load balancer, and configuring virtual host names.	Chapter 7, "Configuring the Web Tier for an Enterprise Deployment"
Create a Domain	Run the Configuration Wizard to create a domain.	Chapter 8, "Creating a Domain for an Enterprise Deployment"
Extend the Domain for SOA	Extend the existing WebLogic domain by running the Configuration Wizard to configure Oracle SOA components.	Chapter 9, "Extending the Domain for SOA Components"
Extend the Domain for BAM?	You have two options for extending the domain to include Oracle BAM components: ■ If you have not yet extended the domain for Oracle SOA, you can run the configuration Wizard and configure Oracle SOA and Oracle BAM at the same time. ■ If you have already extended the domain to include Oracle SOA, you can run the Configuration Wizard again and extend the domain to include Oracle BAM.	Chapter 10, "Extending the Domain to Include Oracle BPM"

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

Step	Description	More Information
Extend the Domain for BAM?	You have two options for extending the domain to include Oracle BAM components: ■ If you have not yet extended the domain for Oracle SOA, you can run the configuration Wizard and configure Oracle SOA and Oracle BAM at the same time. ■ If you have already extended the domain to include Oracle SOA, you can run the Configuration Wizard again and extend the domain to include Oracle BAM.	Chapter 12, "Extending the Domain to Include BAM"
Extend the Domain for OSB?	Run the Configuration Wizard again and extend the domain to include Oracle Service Bus.	Chapter 11, "Extending a SOA Domain to Oracle Service Bus"
Set up Node Manager	Set up Node manager by enabling host name verification, starting Node Manager, and configuring WebLogic Servers to use custom keystores.	Chapter 13, "Setting Up Node Manager for an Enterprise Deployment"
Configure Server Migration	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.	Chapter 14, "Configuring Server Migration for an Enterprise Deployment"
Integrate with Identity Management	You can integrate your Oracle SOA enterprise deployment with Oracle Identity Management 10*g* or 11*g*.	Chapter 15, "Integrating an Enterprise Deployment with Oracle Identity Management"

2.6.3 Understanding the Incremental, Modular Approach to Enterprise Deployment

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

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

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

1. Install the Oracle Fusion Middleware software on disk and create the necessary binary directories.

2. Run the Oracle Fusion Middleware Configuration Wizard to create the domain and configure only the administration components.

The administration components include the Administration Server, Oracle WebLogic Server Administration Console, Oracle Enterprise Manager Fusion Middleware Control, and Oracle Web Services Manager.

3. Run the Configuration Wizard again to extend the domain to include the primary Oracle Fusion Middleware product you want to use.

4. Optionally, run the Configuration Wizard again to extend the domain to include other supporting components and products.

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

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

3

Preparing the Network for an Enterprise Deployment

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

This chapter includes the following topics:

- Section 3.1, "Overview of Preparing the Network for an Enterprise Deployment"
- Section 3.2, "About Virtual Server Names Used by the Topology"
- Section 3.3, "Configuring the Load Balancers"
- Section 3.4, "About IPs and Virtual IPs"
- Section 3.6, "About Firewalls and Ports"

3.1 Overview of Preparing the Network for an Enterprise Deployment

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

You configuring the virtual hosts and associated ports later during Web tier configuration and the installation of the different software components. After reading this chapter, prepare a list of virtual host names and VIPs to be used in this environment.

3.2 About Virtual Server Names Used by the Topology

The SOA enterprise topology uses the following virtual server names:

- soa.mycompany.com
- admin.mycompany.com
- osb.mycompany.com
- soainternal.mycompany.com

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

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

3.2.1 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`.

3.2.2 admin.mycompany.com

`admin.mycompany.com` is a virtual server name that acts as the access point for all internal HTTP traffic that is directed to administration services such as 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.

3.2.3 osb.mycompany.com

`osb.mycompany.com` is a virtual server name that acts as the access point for all HTTP traffic to the runtime Oracle Service Bus resources and proxy services. Traffic to SSL is configured. Clients access this service using the address `osb.mycompany.com:443`.

3.2.4 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.

3.3 Configuring the Load Balancers

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

There are two load balancer devices in the recommended topologies. One load balancer is set up for external HTTP traffic and the other load balancer is set up for internal LDAP traffic. A deployment may choose to have a single load balancer device due to a variety of reasons. While this is supported, the deployment should consider

the security implications of doing this and if found appropriate, open up the relevant firewall ports to allow traffic across the various DMZs. It is worth noting that in either case, it is highly recommended to deploy a given load balancer device in fault tolerant mode.

3.3.1 Load Balancer Requirements

The enterprise topologies use an external load balancer. This external load balancer must have the following features:

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

- Port translation configuration.

- Monitoring of ports (HTTP and HTTPS).

- Virtual servers and port configuration: Ability to configure virtual server names and ports on your external load balancer, and the virtual server names and ports must meet the following requirements:

  - The load balancer should allow configuration of multiple virtual servers. For each virtual server, the load balancer should allow configuration of traffic management on more than one port. For example, for Oracle WebLogic Clusters, the load balancer must be configured with a virtual server and ports for HTTP and HTTPS traffic.

  - The virtual server names must be associated with IP addresses and be part of your DNS. Clients must be able to access the external load balancer through the virtual server names.

- Ability to detect node failures and immediately stop routing traffic to the failed node.

- Resource monitoring / port monitoring / process failure detection: The load balancer must be able to detect service and node failures (through notification or some other means) and to stop directing non-Oracle Net traffic to the failed node. If your external load balancer has the ability to automatically detect failures, you should use it.

- Fault tolerant mode: It is highly recommended that you configure the load balancer to be in fault-tolerant mode.

- Other: It is highly recommended that you configure the load balancer virtual server to return immediately to the calling client when the back-end services to which it forwards traffic are unavailable. This is preferred over the client disconnecting on its own after a timeout based on the TCP/IP settings on the client machine.

- SSL acceleration (this feature is recommended, but not required).

- Configure the virtual server(s) in the load balancer for the directory tier with a high value for the connection timeout for TCP connections. This value should be more than the maximum expected time over which no traffic is expected between Oracle Access Management Access Manager and the directory tier.

- Ability to Preserve the Client IP Addresses: The Load Balancer must have the capability to insert the original client IP address of a request in an X-Forwarded-For HTTP header to preserve the Client IP Address.

3.3.2 Load Balancer Configuration Procedures

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

1. Create a pool of servers. This pool contains a list of servers and the ports that are included in the load balancing definition. For example, for load balancing between the web hosts you create a pool of servers which would direct requests to hosts WEBHOST1 and WEBHOST2 on port 7777.

2. Create rules to determine whether or not a given host and service is available and assign it to the pool of servers described in Step 1.

3. Create a Virtual Server on the load balancer. This is the address and port that receives requests used by the application. For example, to load balance Web Tier requests you would create a virtual host for `sso.mycompany.com:80`.

   a. If your load balancer supports it, specify whether or not the virtual server is available internally, externally or both. Ensure that internal addresses are only resolvable from inside the network.

   b. Configure SSL Termination, if applicable, for the virtual server.

   c. Assign the Pool of servers created in Step 1 to the virtual server.

4. Tune the time out settings as listed in Table 3–4, " Ports Used". This includes time to detect whether a service is down.

3.3.3 Load Balancer Configuration

For an Oracle SOA deployment, configure your load balancer as shown in Table 3–1.

Table 3–1   Load Balancer Configuration

Virtual Host	Server Pool	Protocol	SSL Termination	External	Other Required Configuration/Comments

`admin.myc ompany.co m:80`	`WEBHOST1.m ycompany.c om:7777`
	
	`WEBHOST2.m ycompany.c om:7777`
		HTTP	No	No	■ Use your internal administration address as the virtual server address (for example, `admin.mycompany .com`).
					
					■ 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.

Table 3–1   (Cont.) Load Balancer Configuration

Virtual Host	Server Pool	Protocol	SSL Termination	External	Other Required Configuration/Comments
soa.mycompany.com:443	WEBHOST1.mycompany.com:7777 WEBHOST2.mycompany.com:7777	HTTP	No	Yes	■ 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. ■ Use port 80 and port 443. Any request that goes to port 80 (non-ssl protocol) should be redirected to port 443 (ssl protocol). ■ 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.

Table 3–1   (Cont.) Load Balancer Configuration

Virtual Host	Server Pool	Protocol	SSL Termination	External	Other Required Configuration/Comments
soaintern al.mycomp any.com:8 0	WEBHOST1.m ycompany.c om:7777 WEBHOST2.m ycompany.c om:7777	HTTP	No	No	■ Use your internal administration address as the virtual server address (for example, soainternal.myc ompany.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.
osb.mycom pany.com: 443	WEBHOST1.m ycompany.c om:7777 WEBHOST2.m ycompany.c om:7777	HTTP	No	No	Specify HTTP as the protocol.

3.4 About IPs and Virtual IPs

Configure the Administration Server and the managed servers to listen on different virtual IPs and physical IPs as illustrated in Figure 3–1. As shown in this figure, each virtual IP and IP is attached to the WebLogic server that uses it. VIP1 is failed over 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 3–1   IPs and Virtual IPs Mapped to Administration Server and Managed Servers
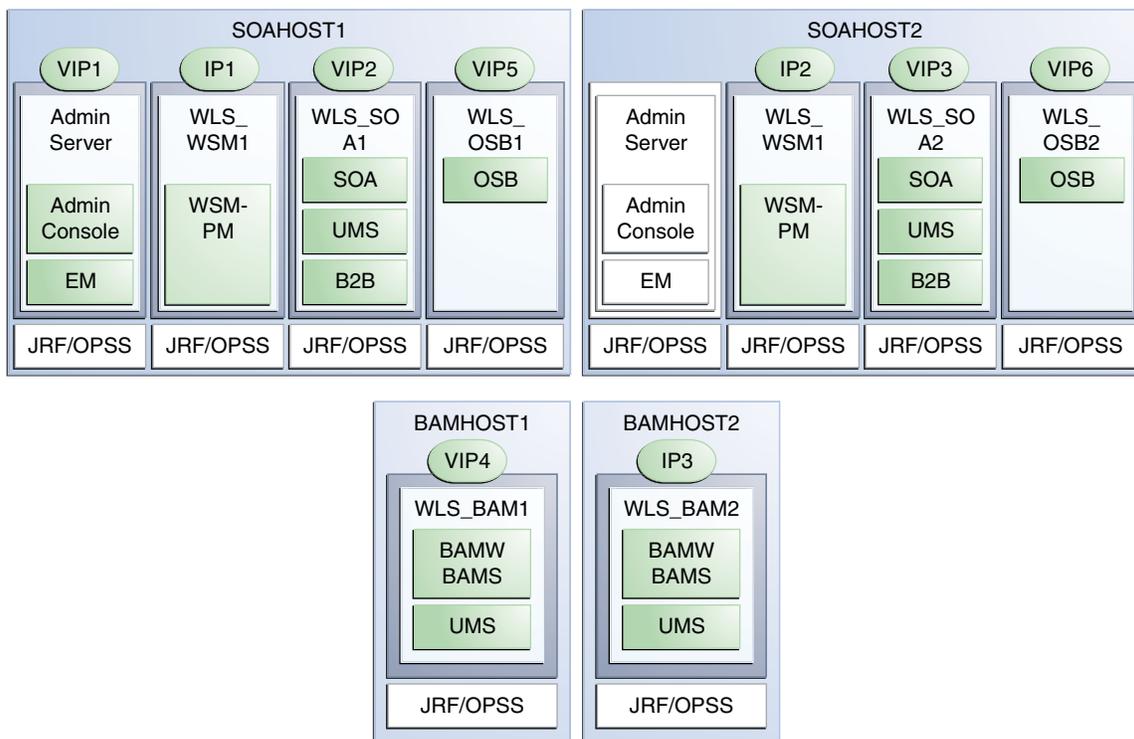
Table 3–2 provides descriptions of the various virtual hosts.

Table 3–2   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).
VIP5	SOAHOST1VHN2	SOAHOST1VHN2 is the virtual host name that maps to the listen address for the WLS_OSB1 server and fails over with server migration of this server. It is enabled in the node where the WLS_OSB1 process us running (SOAHOST1 by default)

Table 3–2 (Cont.) Virtual Hosts

Virtual IP	VIP Maps to...	Description
VIP6	SOAHOST2VHN2	SOAHOST2VHN2 is the virtual host name that maps to the listen address for the WLS_OSB2 server and fails over with server migration of this server. It is enabled in the node where the WLS_OSB2 process us running (SOAHOST2 by default)

3.5 Enabling Virtual IP Addresses for Administration Servers

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

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

Check that the virtual host is enabled as follows:

Table 3–3 Virtual Hosts

VIP	Enabled on Host
ADMINVHN.mycompany.com	SOAHOST1
SOAHOST1VHN1.mycompany.com	SOAHOST1
SOAHOST2VHN1.mycompany.com	SOAHOST2
BAMHOST1VHN1.mycompany.com	BAMHOST1
SOAHOST1VHN2.mycompany.com	SOAHOST1
SOAHOST2VHN2.mycompany.com	SOAHOST2

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

Linux

To enable the virtual IP address, run the following commands as `root`:

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

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

For example:

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

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

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

Validate that the address is available by `ping`ing it from another node, for example:

```
/bin/ping 100.200.140.206
```

Windows

To enable the virtual IP address, run the following command:

```
netsh interface ip add address interface IP_Address netmask
```

where *IP_Address* is the virtual IP address and the *netmask* is the associated netmask.

In the following example, the IP address is enabled on the interface `Local Area Connection`.

```
netsh interface ip add address "Local Area connection" 100.200.140.206
255.255.255.0
```

3.6 About Firewalls and Ports

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

Most port numbers are assigned during installation.

Table 3–4 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 3–4   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.

Table 3–4   (Cont.) Ports Used

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
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 3.3, "Configuring the Load Balancers."
OHS registration with Administration Server	FW1	7001	HTTP/t3	Inbound	Set the timeout to a short period (5-10 seconds).
OHS management by Administration Server	FW1	OPMN port (6701) and OHS Admin Port (7779)	TCP and HTTP, respectively	Outbound	Set the timeout to a short period (5-10 seconds).
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 - 8010	HTTP / WLS_SOA*n*	Inbound	Timeout varies based on the type of process model used for SOA.
Oracle Service Bus Access	FW1	8011 Range: 8011-8021	HTTP / WLS_OSBn	Inbound/ Outbound	Set the timeout to a short period (5-10 seconds).
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.

⚙

Table 3–4   (Cont.) Ports Used

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
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 (OAM 10*g*) 5575 (OAM 11*g*)	OAP	Inbound	For actual values, see "Firewalls and Ports" in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*.
Identity Server access (OAM 10*g*)	FW1	6022	OAP	Inbound	n/a
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.

Table 3–4   (Cont.) Ports Used

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
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
Oracle Notification Server (ONS)	FW2	6200	ONS	Both	Required for Gridlink. An ONS server runs on each database server.

> **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.

4

Preparing the File System for an Enterprise Deployment

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

This chapter includes the following topics:

- Section 4.1, "Overview of Preparing the File System for Enterprise Deployment"
- Section 4.2, "Terminology for Directories and Directory Environment Variables"
- Section 4.3, "About Recommended Locations for the Different Directories"
- Section 4.4, "Configuring Shared Storage"

4.1 Overview of Preparing the File System for Enterprise Deployment

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

Use this chapter as a reference to help understand the directory variables used in the installation and configuration procedures. Other directory layouts are possible and supported, but the model adopted in this guide is chosen for maximum availability, providing both the best isolation of components and symmetry in the configuration and facilitating backup and disaster recovery. The rest of the document uses this directory structure and directory terminology.

4.2 Terminology for Directories and Directory Environment Variables

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

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

- **ORACLE_HOME:** This environment variable and related directory path refer to the location where Oracle Fusion Middleware SOA Suite is installed.

- **ORACLE_COMMON_HOME**: This environment variable and related directory path refer to the Oracle home that contains the binary and library files required for the Oracle Enterprise Manager Fusion Middleware Control and Java Required Files (JRF).

- **ORACLE_HOME_OSB**: This environment variable and related directory path refer to the location where Oracle Service Bus is installed.

- **DOMAIN Directory**: This directory path refers to the location where the WebLogic Server Domain information (configuration artifacts) is stored. Different Oracle WebLogic Servers can use different domain directories even when in the same node as described 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.

- **JAVA_HOME**: This is the location where JRockit is installed.

- **ASERVER_HOME**: This is the primary location of the domain configuration. A typical example is: `/u01/oracle/config/domains/domain_name`

- **MSERVER_HOME**: This is a copy of the domain configuration used to start and stop managed servers. A typical example is: `/u02/private/oracle/config/domains/domain_name`

- **WEB_ORACLE_HOME**: This variable contains the binary and library files required for Oracle HTTP Server and is located in `MW_HOME/web`.

- **WEBGATE_ORACLE_HOME**: This is the location of the WebGate installation.

  > **Tip:** You can simplify directory navigation by using environment variables as shortcuts to the locations in this section. For example, you could use an environment variable called $ORACLE_BASE in Linux to refer to /u01/app/oracle (that is, the recommended ORACLE_ BASE location). In Windows, you would use %ORACLE_BASE% and use Windows-specific commands.

4.3 About Recommended Locations for the Different Directories

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

- Shared Storage Recommendations for Binary (Oracle Home) Directories

- Shared Storage Recommendations for Domain Configuration Files

- Shared Storage Recommendations for JMS File Stores and Transaction Logs

- Recommended Directory Locations

- Directory Structure and Configurations

4.3.1 Shared Storage Recommendations for Binary (Oracle Home) Directories

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

- About the Binary (Oracle Home) Directories

- [About Sharing a Single Oracle Home for Multiple Domains](#)
- [About Using Redundant Binary (Oracle Home) Directories](#)

4.3.1.1 About the Binary (Oracle Home) Directories

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

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

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

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

4.3.1.2 About Sharing a Single Oracle Home for Multiple Domains

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

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

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

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

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

4.3.1.3 About Using Redundant Binary (Oracle Home) Directories

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

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

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

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

4.3.2 Shared Storage Recommendations for Domain Configuration Files

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

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

- Local Storage Requirements for Managed Server Domain Configuration Files

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

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

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

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

4.3.2.2 Local Storage Requirements for Managed Server Domain Configuration Files

Managed Server configuration Files should reside on local storage to prevent performance issues associated with contention. The directory is known as the MSERVER_HOME directory. It is highly recommended that managed server domain configuration files be placed onto local storage.

If you must use shared storage, it is recommended that you create a storage partition for each node and mount that storage exclusively to that node.

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

4.3.3 Shared Storage Recommendations for JMS File Stores and Transaction Logs

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

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

4.3.4 Recommended Directory Locations

This section describes 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

Recommended directory: /u01/app/oracle

Domain Directory for Administration Server Domain Directory:

ORACLE_BASE/admin/*domain_name*/aserver/*domain_name* (The last "domain_ name" is added by Configuration Wizard)

- Mount point on machine: *ORACLE_BASE*/admin/*domain_name*/aserver

- Shared storage location: *ORACLE_BASE*/admin/*domain_name*/aserver

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

Domain Directory for Managed Server 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

This directory is local in the context of a SOA enterprise deployment.

MW_HOME (application tier)

Recommended directory: *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)

ORACLE_HOME (web tier)

Recommended directory: *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

Recommended directory: *MW_HOME*/wlserver_10.3

ORACLE_HOME

Recommended directory: *MW_HOME*/soa

ORACLE_COMMON_HOME

Recommended directory: *MW_HOME*/oracle_common

ORACLE_HOME_OSB

Recommended directory: */MW_HOME*/osb

ORACLE_INSTANCE (OHS Instance)

Recommended directory: *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)`.

4.3.5 Directory Structure and Configurations

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

Figure 4–1 illustrates the directory structure

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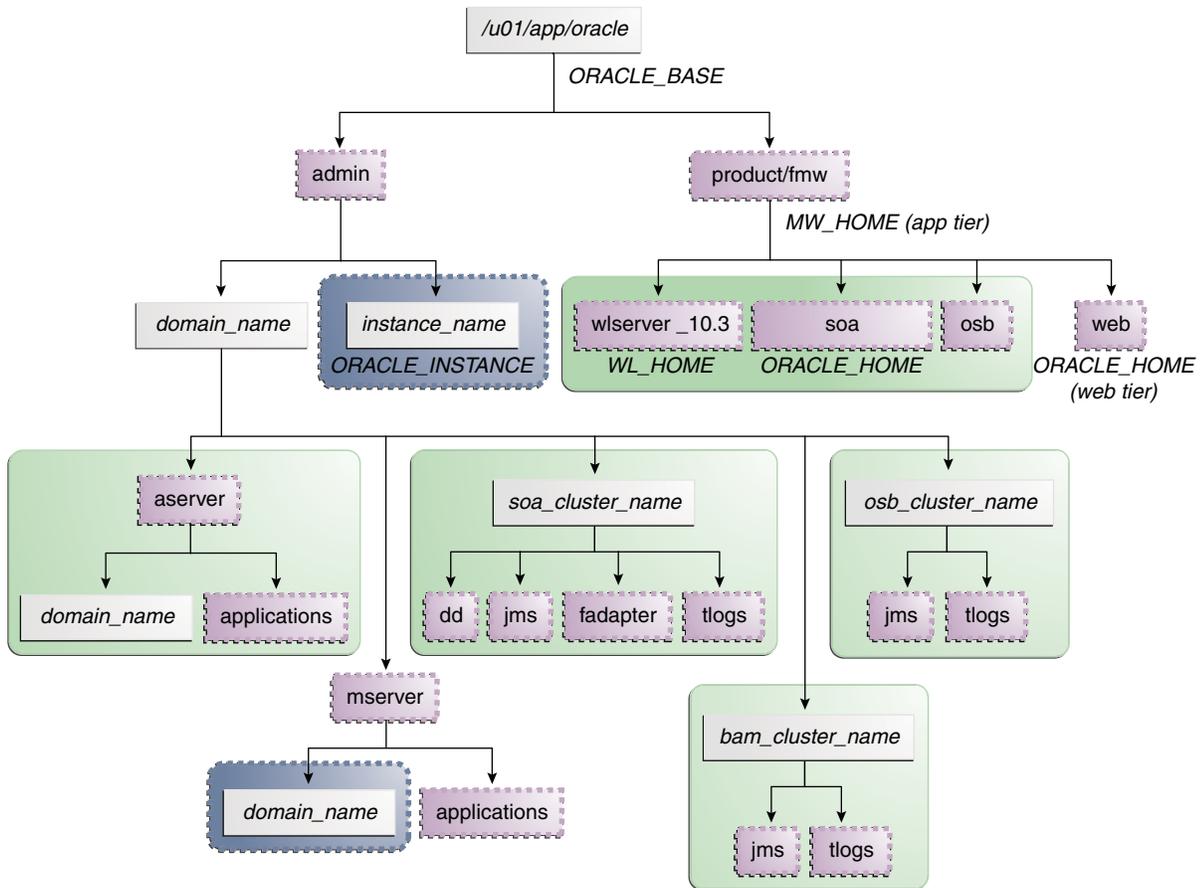
The directory structure in Figure 4–1 does not show other required internal directories, such as `oracle_common` and `jrockit`.

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

Table 4–1   Directory Structure Elements

Element	Explanation
	The Administration Server domain directories, applications, deployment plans, file adapter control directory, JMS and TX logs, and the entire 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 4–2 shows an example configuration for shared storage with multiple volumes for SOA. This can be extrapolated with the same structure for BAM deployments.

Figure 4–2 Example Configuration for Shared Storage
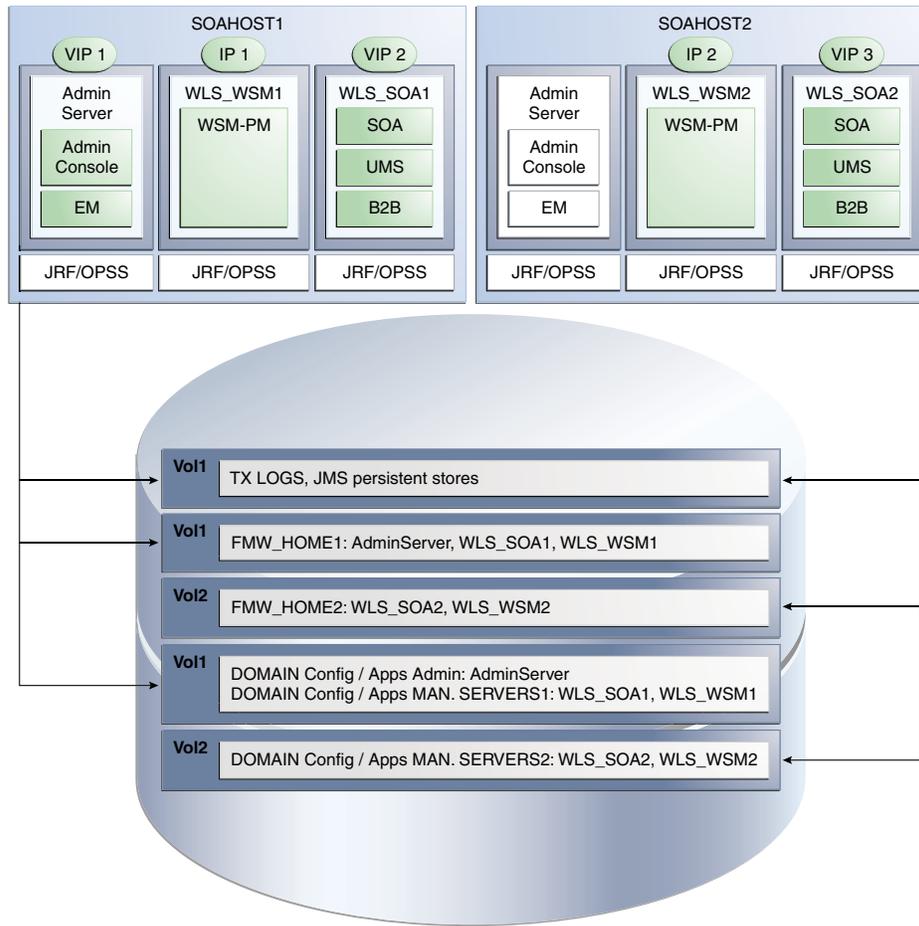
Table 4–2 summarizes the directory structure for the domain.

Table 4–2 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 4–2   (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.
WLS_SOA1	Domain Config	VOL1	*ORACLE_BASE*/admin/*domain_name*/aserver/*domain_name*	Used by only one Server where the Administration server is running.

4.4 Configuring Shared Storage

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

> **Note:**   The user ID used to create a shared storage file system owns 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 the "Understanding Installation and Configuration Privileges and Users" section in the *Oracle Fusion Middleware Installation Planning Guide*.

"nasfiler" is the shared storage filer.

From SOAHOST1:

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

From SOAHOST2:

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

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

From SOAHOST1:

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

From SOAHOST2:

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

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

```
mount nasfiler:/vol/vol1/ORACLE_BASE/stores/soadomain/soa_cluster/tlogs
/ORACLE_BASE/stores/soadomain/soa_cluster/tlogs -t nfs
```

```
mount nasfiler:/vol/vol1/ORACLE_BASE/stores/soadomain/soa_cluster/tlogs
/ORACLE_BASE/stores/soadomain/soa_cluster/tlogs -t nfs
```

Validating the Shared Storage Configuration

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

For example:

```
$ cd newly mounted directory
$ touch testfile
```

Verify that the owner and permissions are correct:

```
$ ls -l testfile
```

Then remove the file:

```
$ rm testfile
```

> **Note:** The shared storage can be a NAS or SAN device. The following illustrates an example of creating storage for a NAS device from SOAHOST1. The options may differ depending on the specific storage device.
>
> ```
> mount nasfiler:/vol/vol1/fmw11shared ORACLE_BASE/wls -t nfs -o
> rw,bg,hard,nointr,tcp,vers=3,timeo=300,rsize=32768, wsize=32768
> ```
>
> Contact your storage vendor and machine administrator for the correct options for your environment.

5

Preparing the Database for an Enterprise Deployment

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

This chapter includes the following topics:

- Section 5.1, "Overview of Preparing the Database for an Enterprise Deployment"
- Section 5.2, "About Database Requirements"
- Section 5.3, "Creating Database Services"
- Section 5.4, "Loading the Oracle Fusion Metadata Repository in the Oracle RAC Database"
- Section 5.5, "Configuring SOA Schemas for Transactional Recovery Privileges"
- Section 5.6, "Backing Up the Database"

5.1 Overview of Preparing the Database for an Enterprise Deployment

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 2–2. 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.

5.2 About Database Requirements

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

- Section 5.2.1, "Database Host Requirements"

- Section 5.2.2, "Supported Database Versions"

- Section 5.2.3, "About Initialization Parameters"

5.2.1 Database Host Requirements

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

- **Oracle Clusterware**

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

- **Oracle Real Application Clusters**

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

- **Automatic Storage Management** (optional)

    ASM is installed for the node as a whole. Oracle recommends installing it in a separate Oracle Home from the Database Oracle Home. This option comes appears in the Select Configuration page. Select the Configure Automatic Storage Management option to create a separate ASM home.

- **Time synchronization between Oracle RAC database instances**

    The clocks of the database instances being used by servers in a Fusion Middleware cluster that is configured with server migration must be in sync.

5.2.2 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 "Oracle Fusion Middleware 11*g* Release 1 (11.1.1.x)" product area on the Oracle Fusion Middleware Supported System Configurations page:

    http://www.oracle.com/technology/software/products/ias/files/fusion_certification.html

To check the release of your database query the PRODUCT_COMPONENT_VERSION view:

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

Notes:

- Oracle SOA requires that the database used to store its metadata (either 10*g* or 11*g*) supports the AL32UTF8 character set. Check the database documentation for information on choosing a character set for the database.

- For Oracle SOA enterprise deployments, Oracle recommends using GridLink data sources to connect to Oracle RAC databases. To use the Oracle Single Client Access Name (SCAN) feature with GridLink, the Oracle RAC database version must be Oracle Database 11*g*R2 (11.2 or later, Enterprise Edition).

5.2.3 About Initialization Parameters

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

Table 5–1   Required Initialization Parameters

Configuration	Parameter	Required Value	Parameter Class
SOA	PROCESSES	300 or greater	Static
BAM	PROCESSES	100 or greater	Static
SOA and BAM	PROCESSES	400 or greater	Static
SOA and OSB	PROCESSES	800 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.

5.3 Creating Database Services

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

Automatic Workload Management" in the *Oracle Real Application Clusters Administration and Deployment Guide*.

Run-time connection load balancing requires configuring Oracle RAC Load Balancing Advisory with service-level goals for each service for which load balancing is enabled. You can configure the Oracle RAC Load Balancing Advisory for `SERVICE_TIME` or `THROUGHPUT`. Set the connection load balancing goal to **SHORT**. For 10*g* and 11*g*r1 databases, use the `DBMS_SERVICE` package for this modification. For 11*g* R2 use the `srvctl` command utility instead.

This section includes the following topics:

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

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

You can create and modify 10*g* and 11*g* database services using the `DBMS_SERVICE` package.

To create and modify database services:

1. Logon to SQL*Plus and create the service:

```
SQL*Plus "sys/password as sysdba"

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

> **Note:** For the Service Name of the Oracle RAC database, use lowercase letters, followed by the domain name. For example:
>
> `soaedg.mycompany.com`

> **Note:** Enter the `EXECUTE DBMS_SERVICE` command shown on a single line.
>
> For more information about the `DBMS_SERVICE` package, see the *Oracle Database PL/SQL Packages and Types Reference*.

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

```
srvctl add service -d soadb -s soaedg.mycompany.com -r soadb1,soadb2
```

3. Start the service:

```
srvctl start service -d soadb -s soaedg.mycompany.com
```

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

4. Modify the service for the appropriate service goals:

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

Or

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

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

You can create and modify 11*g* Release 2 (11.2) database services using the srvctl utility.

To create and modify the database services:

1. Logon to SQL*Plus and create the service:

```
sqlplus "sys/password as sysdba"

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

> **Note:** For the Service Name of the Oracle RAC database, use lowercase letters, followed by the domain name. For example:
>
> ```
> soaedg.mycompany.com
> ```

> **Note:** Enter the EXECUTE DBMS_SERVICE command shown on a single line.
>
> For more information about the DBMS_SERVICE package, see the *Oracle Database PL/SQL Packages and Types Reference*.

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

```
srvctl add service -d soadb -s soaedg.mycompany.com -r soadb1,soadb2
```

3. Start the service:

```
srvctl start service -d soadb -s soaedg.mycompany.com
```

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

4. Modify the service for the appropriate service goals:

   ```
   srvctl modify service -d soadb -s soaedg.mycompany.com -B SERVICE_TIME -j SHORT
   ```

   Or

   ```
   srvctl modify service -d soadb -s soaedg.mycompany.com -B  THROUGHPUT  -j SHORT
   ```

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

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

The Repository Creation Utility (RCU) is available from the RCU DVD. The RCU used to seed the database must match the patch set level of the Oracle SOA Suite installation. This means that if you install Oracle SOA Suite 11*g*R1 PS5 (11.1.1.6) in this enterprise deployment, you must use RCU 11*g*R1 PS5 (11.1.1.6).

To load the Oracle Fusion Middleware Repository into a database:

1. Start Repository Creation Utility (RCU), which is available from the RCU DVD by first inserting the RCU DVD.

2. Start RCU from the *bin* directory:

   ./rcu

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

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

5. In the Database Connection Details screen, enter the correct information for your database:

   a. **Database Type**: select **Oracle Database**.

   b. **Host Name**: Enter the name of the node that is running the database. For the Oracle RAC database, specify the virtual IP name or one of the node names as the host name: CUSTDBHOST1-VIP.

   > **Note:** You can use the RAC SCAN address as the host name. For more information about using SCAN addresses, "Using SCAN Addresses with Oracle Database 11g (11.2)" in the *Oracle Fusion Middleware High Availability Guide*.

   c. **Port**: Enter the port number for the database: 1521.

   d. **Service Name**: Enter the service name of the database in lowercase characters. For example:

```
soaedg.mycompany.com
```

  e. **Username**: SYS

  f. **Password**: Enter the password for the SYS user.

  g. **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:

  a. 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*.

  b. Note the name of the schema because you will need to enter it during the procedure in Section 5.5.

  c. Select the following:

    – AS Common Schemas:

      - **Metadata Services**

    – SOA and BPM Infrastructure:

      - **SOA Infrastructure**

      - **User Messaging Service**

      - **Business Activity Monitoring**

Note: Business Activity Monitoring (BAM) is only required for BAM installations.

Note: Oracle Service Bus required objects are created as part of the SOA_INFRA schema.

  Click **Next**.

8. In the Schema Passwords screen, select **Use main schema passwords for auxiliary schemas**. In the subsequent screen refresh, enter the schema passwords for all components.

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

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

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

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

12. Verify that the required schemas are created by connecting to the database with the new user added:

```
sqlplus  PROD_SOAINFRA/password;
```

Query the description of the CUBE_INSTANCE table for a simple verification. A table similar to the following should display:

```
SQL> desc CUBE_INSTANCE;
 Name                                 Null?     Type
 -------------------------------- ---- -------------------------
 CIKEY                                NOT NULL NUMBER(38)
 CREATION_DATE                        NOT NULL TIMESTAMP(6)
...
```

About Oracle WSM policies and the OWSM MDS schemas

If Oracle WSM is part of your SOA enterprise deployment, Oracle recommends using the identity management database to store the Oracle WSM policies. Use the IM database connection information for the OWSM MDS schemas instead of the information 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 7). See Chapter 15, "Integrating an Enterprise Deployment with Oracle Identity Management." for information on using the identity management database to store the Oracle WSM policies.

5.5 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.

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

To configure the SOA schemas for transactional recovery privileges:

1. Log on to SQL*Plus 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_
soainfra;

Grant succeeded.

SQL> Grant force any transaction to soa_schema_prefix_soainfra;

Grant succeeded.

SQL>
```

5.6 Backing Up the Database

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

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

Database Growth Management Strategy

An Oracle SOA Suite 11*g* installation presents several challenges for database administrators, including managing the growth of the Oracle SOA Suite database. Underestimating the importance of managing the database can lead to issues when the database is moved to a production environment. For information about determining an appropriate strategy and planning for capacity, testing, and monitoring, see "Introduction to Planning for Database Growth" in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA and Oracle Business Process Management Suite*.

6

Installing the Software for an Enterprise Deployment

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

This chapter contains the following sections:

- Section 6.1, "Overview of the Software Installation Process"
- Section 6.2, "Installing Oracle HTTP Server"
- Section 6.3, "Installing Oracle Fusion Middleware"

6.1 Overview of the Software Installation Process

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

Obtaining the Software

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

Select one of the download locations and download "SOA Suite." The .zip archive file is saved to your system.

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

Software to Install

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

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

Hosts	Oracle HTTP Server	Oracle WebLogic Server	Oracle SOA Suite
WEBHOST1	X		
WEBHOST2	X		
SOAHOST1		X	X

Table 6–1   (Cont.) Software To Be Installed On Each Host or Accessible From Each Host

Hosts	Oracle HTTP Server	Oracle WebLogic Server	Oracle SOA Suite
SOAHOST2		X	X

6.2 Installing Oracle HTTP Server

This section covers these topics:

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

6.2.1 Prerequisites to Installing Oracle HTTP Server

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

- Ensure that the system, patch, kernel, and other requirements are met as specified in the *Oracle Fusion Middleware Installation Guide for Oracle Web Tier*.

- Because Oracle HTTP Server is installed on port 7777 by default, you must make sure that port 7777 is not used by any service on the nodes. To check if this port is in use, run the following command before installing Oracle HTTP Server:

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

  You must free port 7777 if it is in use.

- On Linux platforms, if the /etc/oraInst.loc file exists, check that its contents are correct. Specifically, check that the inventory directory is correct and that you have write permissions for that directory. If the /etc/oraInst.loc file does not exist, you can skip this step.

- Before starting the installation, make sure that the following environment variables are not set:
  - LD_ASSUME_KERNEL
  - ORACLE_INSTANCE

6.2.2 Installing Oracle HTTP Server on WEBHOST1 and WEBHOST2

When you install Oracle HTTP Server, you are installing the Web Tier and NOT associating it with a domain. However, you do need to create a MW_HOME directory for the Web tier, even though it is not associated with a domain.

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

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

   ```
   ./runInstaller
   ```

2. In the Specify Inventory Directory screen, do the following:

   a. Enter *HOME*/**oraInventory**, where *HOME* is the home directory of the user performing the installation (this is the recommended location).

   b. Enter the OS group for the user performing the installation.

    c. Click **OK**.

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

Click **OK**.

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

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

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

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

7. In the Specify Installation Location screen, create a MW_HOME that will contain
your Web Tier home directory. You are creating two new directories. You are not
selecting an existing directory. For this screen, enter the following:

    ■ **Fusion Middleware Home Location** (installation location): *ORACLE_
BASE*/product/fmw

    ■ **Oracle Home Directory:** web

Click **Next**.

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

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

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

The Oracle HTTP Server software is installed.

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

6.2.3 Backing Up the Oracle Fusion Middleware Installation

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

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

6.3 Installing Oracle Fusion Middleware

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

> **Note:** Before starting the setup process, read the release notes for additional installation and deployment information. They are available on the Oracle Fusion Middleware Documentation Library at http://docs.oracle.com/cd/E28280_01/relnotes.htm.

This section covers these topics:

- Section 6.3.1, "Installing Oracle WebLogic Server and Creating the Fusion Middleware Home"

- Section 6.3.2, "Installing Oracle Fusion Middleware SOA Suite"

- Section 6.3.3, "Installing the Required Oracle Service Bus Binaries"

- Section 6.3.4, "Backing Up the Fusion Middleware Installation"

6.3.1 Installing Oracle WebLogic Server and Creating the Fusion Middleware Home

To install Oracle WebLogic Server on SOAHOST1 and SOAHOST2:

> **Note:** If you are installing WebLogic Server on a 64-bit platform using a 64-bit JDK, follow the steps in section "Installing WebLogic Server on 64-Bit Platforms Using a 64-Bit JDK" in the *Oracle Fusion Middleware Installation Guide for Oracle WebLogic Server* instead of the steps in this section.

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

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

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

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

   - Select **Create a new Middleware Home**.

   - For Middleware Home Directory, enter *ORACLE_BASE*/**product/fmw**

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

   Click **Next**.

4. In the Register for Security Updates screen, enter your contact information so that you can be notified of security updates, and click **Next**.

5. In the Choose Install Type screen, select **Custom**, and click **Next**.

6. In the Choose Products and Components screen, click **Next**.

7. In the JDK Selection screen, select *only* **Oracle JRockit 1.6.0_<*version*> SDK**, and click **Next**.

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

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

   The Oracle WebLogic Server software is installed.

10. In the Installation Complete screen, clear the **Run Quickstart** check box and click **Done**.

11. Validate the installation by verifying that the following directories and files appear in the ORACLE_HOME directory after installing Oracle WebLogic Server:

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

6.3.2 Installing Oracle Fusion Middleware SOA Suite

To install Oracle Fusion Middleware SOA Suite on SOAHOST1 and SOAHOST2:

1. On Linux platforms, if the /etc/oraInst.loc file exists, check that its contents are correct. Specifically, check that the inventory directory is correct and that you have write permissions for that directory. If the /etc/oraInst.loc file does not exist, you can skip this step.

2. Start the installer for Oracle Fusion Middleware SOA Suite from Disk 1 of the installation media:

```
./runInstaller
```

   When the installer prompts you for a JRE/JDK location, enter the Oracle SDK location created in the Oracle WebLogic Server installation; for example, *ORACLE_BASE*/product/fmw/jrockit-jdk1.6.0_*version*. For more information, see Section 6.3.1, "Installing Oracle WebLogic Server and Creating the Fusion Middleware Home."

3. In the Specify Inventory Directory screen, do the following:

   a. Enter *HOME***/oraInventory**, where *HOME* is the home directory of the user performing the installation (this is the recommended location).

   b. Enter the OS group for the user performing the installation.

   c. Click **OK**.

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

   Click **OK**.

   > **Note:** The Specify Inventory Directory screen appears only on a UNIX operating system, for the first installation by Oracle Universal Installer. The installer uses the inventory directory to keep track of all Oracle products installed on the machine.

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

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

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

7. In the Specify Installation Location screen, provide the installation location for Oracle Fusion Middleware SOA Suite. Select the previously installed Oracle Middleware Home from the drop-down list. For the Oracle Home directory, enter the directory name (**soa**).

Figure 6–1   Specify Installation Location Screen in Installer Wizard

Click **Next** when you are done.

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

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

   The Oracle Fusion Middleware SOA Suite software is installed.

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

11. Validate the installation by verifying that the following directories and files appear in the ORACLE_HOME directory after installing both Oracle WebLogic Sever and Oracle Fusion Middleware for SOA:

   ■ coherence_*X.X*

   ■ jrockit-jdk*Y.Y*

   ■ modules

   ■ oracle_common

   ■ registry.xml

   ■ utils

- domain-registry.xml

- logs

- ocm.rsp

- registry.dat

- soa

- wlserver_10.3

6.3.3 Installing the Required Oracle Service Bus Binaries

To install Oracle Fusion Middleware Oracle Service Bus 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 Service Bus from the installation media:

    ```
    ./runInstaller
    ```

    When the installer prompts you for a JRE/JDK location, enter the Oracle SDK location created in the Oracle WebLogic Server installation; for example, ORACLE_BASE/product/fmw/jrockit-jdk1.6.0_version. For more information, see Section 6.3.1, "Installing Oracle WebLogic Server and Creating the Fusion Middleware Home."

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

    > **Note:**   Since SOAHOST1 and SOAHOST2 already contain the SOA Suite Oracle Home, an Oracle Inventory should already be present and used by this installation hence the "Specify Inventory Directory screen" should not appear

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

5. In the Installation Location screen, provide the installation location for Oracle Service Bus. Select the previously installed Oracle Middleware Home from the drop-down list. For the Oracle Home directory, enter the directory name (osb). Click **Next**.

6. In the Installation Type, select **Custom** and click **Next**.

7. In the Components to Install screen, DESELECT **Oracle Service Bus IDE** and **Oracle Service Bus Examples** and click **Next**.

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

9. In the Product Home Location specify the WebLogic Server installation directory previously installed and click **Next**.

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

11. In the Product Home Location specify the WebLogic Server installation directory previously installed and click **Next**.

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

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

14. Validate the installation by verifying that the following directories appear in the ORACLE_HOME_OSB directory (under osb) after installing Oracle Service Bus:

- 3rdparty

- Common

- Diagnostics

- Install

- Lib

- oraInst.loc

- soa

- bin

- config

- financial

- inventory

- modules

- osb

- cfgtoollogs

- dbscripts

- harvester

- L10N

- OPatch

- oui

15. Repeat the steps in SOAHOST2

6.3.4 Backing Up the Fusion Middleware Installation

The Fusion Middleware Home should be backed up now (make sure that you stop the servers first). From SOHOAST1:

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

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

7

Configuring the Web Tier for an Enterprise Deployment

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

This chapter contains the following sections:

- Section 7.1, "Overview of Configuring the Web Tier"
- Section 7.2, "Prerequisites for Configuring the Web Tier"
- Section 7.3, "Running the Configuration Wizard to Configure Oracle HTTP Server"
- Section 7.4, "Validating the Configuration"
- Section 7.5, "Configuring the Load Balancer to Route HTTP Requests"
- Section 7.6, "Defining Virtual Hosts"

7.1 Overview of Configuring the Web Tier

Before configuring the Oracle Web Tier software, you need to install it on WEBHOST1 and WEBHOST2, as described in Section 6.2, "Installing Oracle HTTP Server," and define the instance home, instance name, and Oracle HTTP Server component name, as described in Section 7.3, "Running the Configuration Wizard to Configure Oracle HTTP Server."

The last section describes how to define the directives of the `<VirtualHost>` section of the `httpd.conf` file on both OHS servers. You created these virtual host names when you configured the load balancer in Section 3.3, "Configuring the Load Balancers."

7.2 Prerequisites for Configuring the Web Tier

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

7.3 Running the Configuration Wizard to Configure Oracle HTTP Server

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

To configure the Oracle web tier:

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

   ```
   WEBHOST1> cd WEB_ORACLE_HOME/bin
   ```

2. Start the Configuration Wizard:

   ```
   WEBHOST1> ./config.sh
   ```

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

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

   Click **Next**.

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

   - Instance Home Location: *ORACLE_BASE*/admin/web*n*

   - AS Instance Name: web*n*

   - OHS Component Name: ohs*n*

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

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

   Click **Next**.

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

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

   The file will look like this:

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

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

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

   Click **Next**.

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

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

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

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

7.4 Validating the Configuration

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

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

7.5 Configuring the Load Balancer to Route HTTP Requests

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

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

7.6 Defining Virtual Hosts

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

This section contains the following topics:

- Section 7.6.1, "Define the IP Address and Port in the httpd.conf File"
- Section 7.6.2, "Creating .conf Files to Define <VirtualHost> Directives"
- Section 7.6.3, "Validating the Configuration"

7.6.1 Define the IP Address and Port in the httpd.conf File

You are defining name-based virtual servers. That means you have to define the IP address and port that will be used for each virtual host you define. You define the IP address and port once, in the `httpd.conf` file, then you can define the actual virtual host names (and their specific URLs) in the virtual host-specific `.conf` files.

To define the IP address and port, add the following entry in the `httpd.conf` file:

```
NameVirtualHost *:7777
```

7.6.2 Creating .conf Files to Define <VirtualHost> Directives

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

There is an `INCLUDE` statement in the `httpd.conf` that includes all `*.conf` files in the directory. This statement makes it possible to create separate virtual host files for each

component, making it easier to update, maintain, and scale-out the virtual host definitions.

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

- `soa_vh.conf`

- `soainternal_vh.conf`

- `admin_vh.conf`

- `osb_vh.conf` (If you plan to extend the domain for Oracle Service Bus)

Create the new files in the following directory:

ORACLE_BASE/admin/*instance_name*/config/OHS/*component_name*/moduleconf

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

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

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

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

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

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

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

4. If you plan to extend the domain for Oracle Service Bus, create the `osb_vh.conf` file and add the following directive:

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

5. Restart both Oracle HTTP Servers:

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

7.6.3 Validating the Configuration

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

- `https://soa.mycompany.com/index.html`

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

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

- `https://osb.mycompany.com/index.html`

> **Note:** You should be able to access these URLs only through the load balancer. If you attempt to access them from a Web server host (WEBHOST1 or WEBHOST2), then these URLs will result in errors. The URL definitions are valid only for accessing the server through the load balancer and the defined virtual host URLs.

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

8

Creating a Domain for an Enterprise Deployment

This chapter describes how to create a domain using the 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.

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

This chapter contains the following sections:

- Section 8.1, "Overview of Creating a Domain"
- Section 8.2, "Enabling VIP1 in SOAHOST1"
- Section 8.3, "Running the Configuration Wizard on SOAHOST1 to Create a Domain"
- Section 8.4, "Post-Configuration and Verification Tasks"
- Section 8.5, "Propagating the Domain Configuration to SOAHOST2"
- Section 8.6, "Configuring the Java Object Cache for Oracle WSM"
- Section 8.7, "Configuring Oracle HTTP Server for the WebLogic Domain"
- Section 8.8, "Verifying Manual Failover of the Administration Server"
- Section 8.9, "Backing Up the WebLogic Domain Configuration"

8.1 Overview of Creating a Domain

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

Table 8–1   Steps for Creating a WebLogic Domain

Step	Description	More Information
Enabling VIP1 in SOAHOST1	Enable ADMINVHN for the SOAHOST1 hostname.	Section 8.2, "Enabling VIP1 in SOAHOST1"
Create a WebLogic Domain	Run the Configuration Wizard to create WebLogic domain.	Section 8.3, "Running the Configuration Wizard on SOAHOST1 to Create a Domain"
Post-Configuration and Verification Tasks	Follow the instructions for post-configuration and validation tasks.	Section 8.4, "Post-Configuration and Verification Tasks"
Propagate the Domain Configuration to SOAHOST2	Propagate the start scripts and classpath configuration from the Administration Server's domain directory to the managed server domain directory.	Section 8.5, "Propagating the Domain Configuration to SOAHOST2"
Configure the Oracle HTTP Server with the WebLogic domain	Configure the Oracle HTTP Server with the WebLogic domain and validate the configuration.	Section 8.7, "Configuring Oracle HTTP Server for the WebLogic Domain"
Back Up the Domain	Back up the newly configured WebLogic domain.	Section 8.9, "Backing Up the WebLogic Domain Configuration"

Once this domain is created and configured you can extend the domain to include Oracle SOA components, Oracle BAM, or Oracle BAM as described in the next chapters.

8.2 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 virtual IP (VIP1) either by a DNS Server or by a custom /etc/hosts entry. Check that ADMINVHN is available according to your name resolution system, (DNS server, /etc/hosts), in the required nodes in your SOA topology. The virtual IP (VIP1) that is associated to this Virtual Host Name (ADMINVHN) must be enabled in SOAHOST1.

To enable the virtual IP, see Section 3.5, "Enabling Virtual IP Addresses for Administration Servers."

8.3 Running the Configuration Wizard on SOAHOST1 to Create a Domain

Run the Configuration Wizard from the Oracle Common 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.

To create a domain:

1. Ensure that the database where you installed the repository is running. For Oracle RAC databases, all instances should be running, so that the validation check later in the procedure is more reliable.

2. Change directory to the location of the Configuration Wizard. This is within the SOA home directory. From SOAHOST1:

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

3. Start the Oracle Fusion Middleware Configuration Wizard:

    `./config.sh`

4. In the Welcome screen, select **Create a New WebLogic Domain**, and click **Next**.

5. The Select Domain Source screen appears (Figure 8–1).

Figure 8–1   Select Domain Source Screen
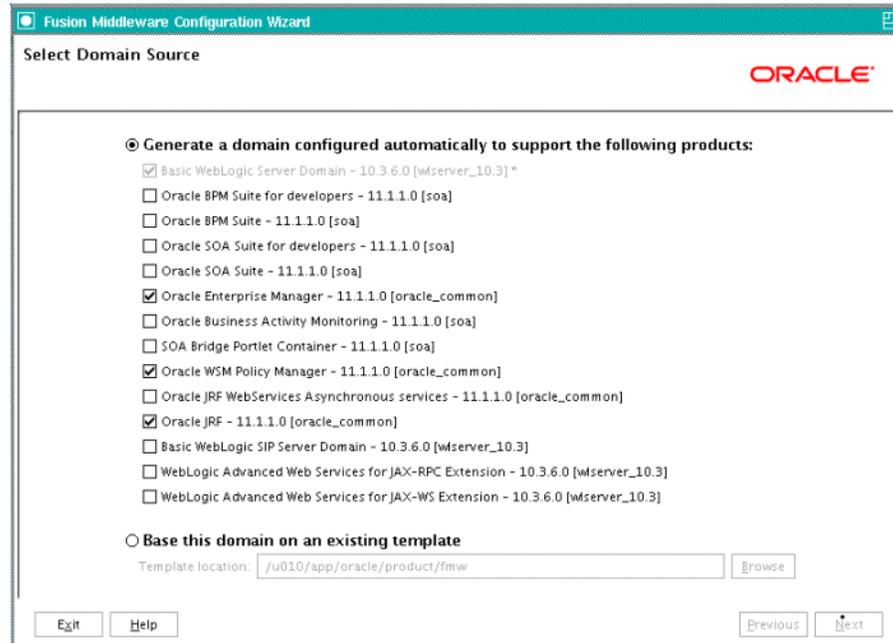
Screenshot of the Select Domain Source screen, where you select the products that the newly created WebLogic domain will support automatically. It is described in further detail in the text following this image.

****

In the Select Domain Source screen, do the following:

■   Select **Generate a domain configured automatically to support the following products**.

■   Select the following products:

    –   **Basic WebLogic Server Domain - 10.3.6.0 [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]**

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 4.3, "About Recommended Locations for the Different Directories." Enter the following for the domain directory:

   `ORACLE_BASE/admin/domain_name/aserver/`

   And the following for the application directory. This directory should be in shared storage:

   `ORACLE_BASE/admin/domain_name/aserver/applications`

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:

    ■ Select the OWSM MDS schema.

    ■ For the Oracle RAC configuration for component schemas, select **Convert to GridLink**.

    Click **Next**.

11. The Configure Gridlink RAC Component Schema screen appears (Figure 8–2).

Figure 8–2   Configure GridLink RAC Component Schema Screen

In this screen enter values for the following fields, specifying the connect information for the Oracle RAC database that was seeded with RCU:

- **Driver:** Select **Oracle's driver (Thin) for GridLinkConnections,Versions:10 and later**.

- **Service Name:** Enter the service name of the database using lowercase characters. For example:

  `soaedg.mycompany.com`.

- **Username:** Enter the database schema owner name of the corresponding component.

- **Password:** Enter the password for the database schema owner.

- Select **Enable FAN**

- Make sure **Enable SSL** is unchecked (alternatively if ssl is selected for ONS notifications to be encrypted, provide the appropriate wallet and wallet password).

- **Service listener**: Enter the SCAN address and port for the RAC database being used. You can identify this address by querying the appropriate parameter in the database using the TCP protocol:

  ```
  SQL>show parameter remote_listener;

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

> **Note:** For Oracle Database 11*g* Release 1 (11.1), use the virtual IP and port of each database's instance listener, for example:
>
> ```
> custdbhost1-vip.mycompany.com (port 1521)
> ```
>
> and
>
> ```
> custdbhost2-vip.mycompany.com (1521)
> ```
>
> For Oracle Database 10*g*, use multi data sources to connect to an Oracle RAC database. For information about configuring multi data sources see Appendix A, "Using Multi Data Sources with Oracle RAC."

- **ONS Host**: Enter the SCAN address for the Oracle RAC database and the ONS remote port as reported by the database:

  ```
  [orcl@db-scan1 ~]$ srvctl config nodeapps -s

  ONS exists: Local port 6100, remote port 6200, EM port 2016
  ```

  > **Note:** For Oracle Database 11*g* Release 1 (11.1), use the hostname and port of each database's ONS service, for example
  >
  > ```
  > custdbhost1.mycompany.com (port 6200)
  > ```
  >
  > and
  >
  > ```
  > custdbhost2.mycompany.com (6200)
  > ```

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

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

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 8–2   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 8–3   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 the **WSM-PM_Cluster** as follows:

- WLS_WSM

- WLS_WSM2

Click **Next**.

18. In the Configure Machines screen, 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 8–4   Machines

Name	Node Manager Listen Address
SOAHOST1	*SOAHOST1*
SOAHOST2	*SOAHOST2*
ADMINHOST	localhost

Leave all other fields to their default values.

> **Note:**   The machine name does not need to be a valid host name or listen address; it is just a unique identifier of a Node Manager location

Click **Next**.

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 is targeted to the **WSM-PM_Cluster** only. Target the library **oracle.wsm.seedpolicies** to **WSM-PM_Cluster**. Make sure that all other deployments are targeted to the **AdminServer** and 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**.

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

   - Click **Next**.

   For information on targeting applications and resources, see Appendix B, "Targeting Applications and Resources to Servers."

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

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

8.4 Post-Configuration and Verification Tasks

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

This section includes the following topics:

- Section 8.4.1, "Creating boot.properties for the Administration Server on SOAHOST1"

- Section 8.4.2, "Starting Node Manager on SOAHOST1"

- Section 8.4.3, "Starting the Administration Server on SOAHOST1"

- Section 8.4.4, "Validating GridLink Data Sources"

- Section 8.4.5, "Validating the Administration Server Configuration"

- Section 8.4.6, "Creating a Separate Domain Directory for Managed Servers in the Same Node as the Administration Server"

- Section 8.4.7, "Applying the Java Required Files (JRF) Template to the WSM-PM_ Cluster"

- Section 8.4.8, "Disabling Host Name Verification"

- Section 8.4.9, "Starting and Validating the WLS_WSM1 Managed Server"

8.4.1 Creating boot.properties for the Administration Server on SOAHOST1

Create a `boot.properties` file for the Administration Server on SOAHOST1. This is a required step that enables you to start the Administration Server using Node Manager.

To create a `boot.properties` file 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 8.4.3, "Starting the Administration Server on SOAHOST1."
   >
   > For security reasons, you want to minimize the time the entries in the file are left unencrypted: after you edit the file, you should start the server as soon as possible so that the entries get encrypted.

3. Save the file and close the editor.

8.4.2 Starting Node Manager on SOAHOST1

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

To start Node Manager on SOAHOST1:

1. Run the `setNMProps.sh` script located in the following directory:

   ```
   ORACLE_COMMON_HOME/common/bin
   ```
   Set the `StartScriptEnabled` property to 'true' before starting Node Manager:

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

   > **Note:** You must use the `StartScriptEnabled` property to avoid class loading failures and other problems. For more information, see also Section 16.14.5, "Incomplete Policy Migration After Failed Restart of SOA Server."

2. Start Node Manager:

   ```
   cd WL_HOME/server/bin
   export JAVA_OPTIONS="-DDomainRegistrationEnabled=true"
   ./startNodeManager.sh
   ```

> **Note:** It is important that you set
> `-DDomainRegistrationEnabled=true` whenever a Node
> Manager that manages the AdminServer is started. If there is no
> AdminServer on this machine and this machine is not an
> AdminServer failover node, you can start the Node Manager using the
> following command from SOAHOAST1:
>
> ```
> ./startNodeManager.sh
> ```

8.4.3 Starting the Administration Server on SOAHOST1

The Administration Server is started and stopped using Node Manager. However, the
first start of the Administration Server with Node Manager, requires changing the
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.

Steps 1-4 are required for the first start operation, subsequent starts require only step 4.

To start the Administration Server using Node Manager:

1. Start the Administration Server using the start script in the domain directory on
   SOAHOST1:

   ```
   cd ASERVER_HOME/bin
   ./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 & Edit**.

   d. Click **domain_name**, **(Security)** tab, **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. Click **Save** and **Activate 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**. Enter the
   Node Manager Username and password that you entered in step 2e.

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

   Once you are in the WLST shell:

   ```
   wls:/offline>nmConnect('nodemanager_username','nodemanager_password',
   'SOAHOST1','5556','domain_name','ASERVER_HOME')

   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 `nm_password.properties` file located in the following directory:
>
> *ORACLE_BASE*/admin/*domain_name*/aserver/*domain_name*/config/nodemanager

8.4.4 Validating GridLink Data Sources

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

To validate the GridLink data sources configuration:

1. Log on to the Oracle WebLogic Administration Console.

2. In the **Domain Structure** tree, expand **Services**, and select **Data Sources**.

3. Click one of the new data sources.

4. Click the **Monitoring** tab and select one of the servers.

5. Click the **Statistics** tab and select one of the servers.

6. Click the **ONS** tab, and then click the **Testing** tab.

7. Select the server and click **Test ONS**.

   If both tests are successful, the configuration is correct. If the ONS test fails, verify that the ONS service is running in the RAC database nodes:

   ```
   orcl@db-scan1 ~]$ srvctl status scan_listener
   SCAN Listener LISTENER_SCAN1 is enabled
   SCAN listener LISTENER_SCAN1 is running on node db-scan1
   SCAN Listener LISTENER_SCAN2 is enabled
   SCAN listener LISTENER_SCAN2 is running on node db-scan2
   SCAN Listener LISTENER_SCAN3 is enabled
   SCAN listener LISTENER_SCAN3 is running on node db-scan2

   [orcl@db-scan1 ~]$ srvctl config nodeapps -s
   ONS exists: Local port 6100, remote port 6200, EM port 2016

   [orcl@db-scan1 ~]$ srvctl status nodeapps | grep ONS
   ONS is enabled
   ONS daemon is running on node: db-scan1
   ONS daemon is running on node: db-scan2
   ```

   Run the ONS test from every WebLogic server that uses the data source.

8.4.5 Validating the Administration Server Configuration

To ensure that the Administration Server for the domain you have created is properly configured, validate the configuration by logging into the Oracle WebLogic Server Administration Console and verifying the managed servers and the cluster are listed, and log into Oracle Enterprise Manager.

To verify that the Administration Server is properly configured:

1. In a browser, go to the following URL:

   ```
   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 the following URL:

   ```
   http://ADMINVHN:7001/em
   ```

6. Log in to EM Console with the username and password you specified in Section 8.4.1, "Creating boot.properties for the Administration Server on SOAHOST1."

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

Use the `pack` and `unpack` commands to separate the domain directory used by the Administration Server from the domain directory used by the managed server in *SOAHOST1* as recommended in Chapter 4, "Preparing the File System for an Enterprise Deployment."

Before running the unpack script, be sure the following directory exists as explained in Section 4.3, "About Recommended Locations for the Different Directories."

ORACLE_BASE/admin/*domain_name*/mserver

To create a separate domain directory:

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

   ```
   cd ORACLE_COMMON_HOME/common/bin

   ./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:

   ```
   cd ORACLE_COMMON_HOME/common/bin

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

   Note: You must have write permissions on the following directory before running the unpack command:

   /ORACLE_BASE/admin/*domain_name*

   For example:

   ORACLE_BASE/admin/soaedg_domain/

> **Note:** The configuration steps provided in this enterprise deployment topology are documented with the assumption that a local (per node) domain directory is used for each managed server.

8.4.7 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 8.4.1, "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.

5. Make sure the JRF template is also applied to the Admin Server. If it is not, the **Apply JRF Template** button shows up in the Enterprise Manager screen for the Admin Server.

For information on targeting applications and resources, see Appendix B, "Targeting Applications and Resources to Servers."

8.4.8 Disabling Host Name Verification

This step is required because you have not set up the appropriate certificates to authenticate the different nodes with the Administration Server (see Chapter 13, "Setting Up Node Manager for an Enterprise Deployment"). Because you have not configured the server certificates, you 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 13, "Setting Up Node Manager for an Enterprise Deployment."

To disable host name verification:

1. Log in to Oracle WebLogic Server Administration Console.

2. Click **Lock & Edit**.

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

4. Click **Servers**. The Summary of Servers page 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 5 to 9for the WLS_WSM1 server.

11. Save and activate the changes.

12. Restart the Administration Server for the changes to take effect.

   To restart the Administration Server:

   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 8.4.3, "Starting the Administration Server on SOAHOST1."

8.4.9 Starting and Validating the WLS_WSM1 Managed Server

After configuring the managed server, start it and check to confirm that it is running properly. You can start the managed server and check its status by using the Oracle WebLogic Server Administration Console.

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 16.14, "Troubleshooting the Topology in an Enterprise Deployment" for possible causes.

3. Access the following URL:

   ```
   http://SOAHOST1:7010/wsm-pm
   ```

4. Click **Validate Policy Manager**.

   If the configuration is correct, a list of policies and assertion templates available in the data store appear. If the configuration is not correct, no policies or assertion templates appear.

8.5 Propagating the Domain Configuration to SOAHOST2

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

This section includes the following topics:

- Section 8.5.1, "Propagating the Domain Configuration to SOAHOST2 Using the unpack Utility"

- Section 8.5.2, "Modify the Upload and Stage Directories to an Absolute Path"

- Section 8.5.3, "Disabling Host Name Verification for the WLS_WSM2 Managed Server"

- Section 8.5.4, "Starting Node Manager on SOAHOST2"

■ Section 8.5.5, "Starting and Validating the WLS_WSM2 Managed Server"

8.5.1 Propagating the Domain Configuration to SOAHOST2 Using the unpack Utility

Propagate the domain configuration using the unpack utility. Before running the unpack script, be sure the following directory exists as explained in Section 4.3, "About Recommended Locations for the Different Directories."

```
ORACLE_BASE/admin/domain_name/mserver
```

To propagate the domain configuration:

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

   ```
   cd ORACLE_COMMON_HOME/common/bin
   scp soadomaintemplate.jar oracle@SOAHOST2:/ORACLE_COMMON_HOME/common/bin
   ```

2. Run the unpack command from the ORACLE_COMMON_HOME/common/bin directory, not from the WL_HOME/common/bin directory on SOAHOST2 to unpack the propagated template.

   ```
   cd ORACLE_COMMON_HOME/common/bin

   ./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 configuration steps provided in this enterprise deployment topology are documented with the assumption that a local (per node) domain directory is used for each managed server.

8.5.2 Modify the Upload and Stage Directories to an Absolute Path

After creating the domain and unpacking to the mserver directory, update the upload and stage directories. These directories are defaulted to:

```
./servers/AdminServer/upload
```

and

```
./servers/server_name/stage
```

As a result, these default directory paths create issues for remote deployments and for deployments using the stage mode.

To avoid these issues, update the upload directory to:

```
MSERVER_HOME/servers/AdminServer/upload
```

And update the stage directory to:

```
MSERVER_HOME/servers/server_name/stage
```

Update these directory paths for all servers.

To update these directories:

1. Access the Administration Console.

2. In the left navigation tree, expand **Domain**, and then **Environment**.

3. Click **Lock & Edit**.

4. Click on **Servers**, and then the server's name.

5. Under the **Configuration**, and then **Deployment** section, change the **Upload** and **Stage** directories.

8.5.3 Disabling Host Name Verification for the WLS_WSM2 Managed Server

For the enterprise deployment described in this guide, you set up the appropriate certificates to authenticate the different nodes with the Administration Server after you have completed the procedures to extend the domain for Oracle SOA Suite. You must disable the host name verification for the WLS_SOA1 and WLS_SOA2 managed servers to avoid errors when managing the different WebLogic Server instances. For more information, see Section 8.4.8, "Disabling Host Name Verification."

You enable host name verification again once the enterprise deployment topology configuration is complete. For more information, see Section 13.3, "Enabling Host Name Verification Certificates for Node Manager in SOAHOST1."

8.5.4 Starting Node Manager on SOAHOST2

Once you have propagated the domain configuration and disabled host name verification, start Node Manager using the StartNodeManager.sh script.

You must use the `StartScriptEnabled` property to avoid class loading failures and other problems. See also Section 16.14.5, "Incomplete Policy Migration After Failed Restart of SOA Server."

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:

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

2. Start Node Manager:

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

8.5.5 Starting and Validating the WLS_WSM2 Managed Server

Use the Administration Console to start and validate the WLS_WSM2 managed server.

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 16.14, "Troubleshooting the Topology in an Enterprise Deployment" for possible causes.

3. Access the following URL:

```
http://SOAHOST2:7010/wsm-pm
```

    4. Click validate policy manager.

8.6 Configuring the Java Object Cache for Oracle WSM

Configure the Java Object Cache (JOC) among all the servers running Oracle WSM. This procedure is optional, but increases the performance of Oracle WSM by keeping a local cache instead of having to search for a cache.

Use JOC for MDS updates in B2B if you are planning to change the delivery channels for B2B agreements frequently.

Configure the Java Object Cache using the configure-joc.py script in the following directory:

```
MW_HOME/oracle_common/bin/
```

This is a Python script that runs in WLST online mode and expects the Administration Server to be up and running.

Use ports in the 9988 to 9998 range when configuring JOC ports for Oracle products.

To configuring the Java Object Cache for Oracle WSM:

1. Connect to the Administration Server on *SOAHOST1* using the command-line Oracle WebLogic Scripting Tool (WLST), for example:

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

When prompted, enter the server URL (t3://ADMINVHN:7001) and the Oracle WebLogic Server administrator user name and password.

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
```

4. After configuring the Java Object Cache using the `wlst` commands or
configure-joc.py script, restart all affected managed servers for the configurations to
take effect.

The script can also be used to perform the following optional 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*.

8.7 Configuring Oracle HTTP Server for the WebLogic Domain

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

This section includes the following topics:

8.7.1 Configuring Oracle HTTP Server for the Administration Server and the WLS_WSM*n* Managed Servers

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

To set the WebLogicCluster parameter:

1. On WEBHOST1 and WEBHOST2, add directives to the `admin_vh.conf` and `soainternal_vh.conf` files located in the following directory:

   ORACLE_BASE/admin/*instance_name*/config/OHS/*component_name*/moduleconf

   Note that this assumes you created the `admin_vh.conf` and `soainternal_vh.conf` files using the instructions in Section 7.6, "Defining Virtual Hosts."

   Add the following directives to the `admin_vh.conf` file within the `<VirtualHost>` tags.

   ```
   # 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>
   ```

   The `admin_vh.conf` file will appear as it does in Example 8–1.

2. Add the following directives to the `soainternal_vh.conf` file within the `<VirtualHost>` tags:

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

   The `soainternal_vh.conf` file will appear as it does in Example 8–2.

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

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

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

Example 8–1   admin_vh.conf file

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

<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>
```

Example 8–2   soainternal_vh.conf file

```
<VirtualHost *:7777>
    ServerName soainternal.mycompany.com:80
    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>
</VirtualHost>
```

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.

8.7.2 Turning on the WebLogic Plug-In enabled Flag

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

To turn on the WebLogic plug-in enabled flag:

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 & Edit**.

5. Select the **WebLogic Plugin Enabled** check box.

6. Save and activate the changes.

8.7.3 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 advantage of doing this is that the Oracle Web Tier can be managed and monitored using the Oracle Enterprise Manager Fusion Middleware Control.

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.

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

8.7.4 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.

The Oracle WebLogic Server Administration Console application tracks changes made to ports, channels and security using the console. When changes made through the

console are activated, the console validates its current listen address, port and protocol. If the listen address, port and protocol are still valid, the console redirects the HTTP request replacing the host and port information with the Administration Server's listen address and port.

To change the Administration Server's frontend URL:

1. Log in to Oracle WebLogic Server Administration Console.

2. Click **Lock & Edit**.

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

4. Click **Servers** to open the Summary of Servers page.

5. Select **Admin Server** in the **Names** column of the table. The Settings page for AdminServer(admin) appears.

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.

10. Disable 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.

    a. Log in to the Oracle WebLogic Server Administration Console.

    b. Click the **preferences** link in the banner.

    c. Click the **shared preferences** tab.

    d. Deselect the **follow configuration changes** check box.

Note: If you have any issues activating any configuration changes after modifying the Frontend Host and Port settings, then refer to Section 16.14.14, "Redirecting of Users to Login Screen After Activating Changes in Administration Console."

8.7.5 Validating Access Through Oracle HTTP Server

To validate 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 16.14, "Troubleshooting the Topology in an Enterprise Deployment" for possible causes.

Validate WSM-PM_Cluster through both Oracle HTTP Server using the following URLs:

- `http://WEBHOST1:7777/wsm-pm`

- `http://WEBHOST2:7777/wsm-pm`

- `http://WEBHOST1:7777/console`

- `http://WEBHOST2:7777/console`

- `http://WEBHOST1:7777/em`

- `http://WEBHOST2:7777/em`

- `https://soa.mycompany.com/wsm-pm`

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

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

After setting the frontend URL to the load balancer address, access to the console through the WEBHOSTn addresses will be redirected by the console to the frontend URL, thus validating the correct configuration of both Oracle HTTP Server and the load balancer.

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

After the registering Oracle HTTP Server as described in Section 8.7.3, "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.

8.8 Verifying Manual Failover of the Administration Server

After configuring the domain, test failover by following the steps in Section 16.7, "Verifying Manual Failover of the Administration Server."

8.9 Backing Up the WebLogic Domain Configuration

Perform a backup to save your domain configuration. Make sure you stop the server first. The configuration files are located in the following directory:

ORACLE_BASE/admin/*domain_name*

To back up the domain configuration run the following command on 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`

9

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 8, "Creating a Domain for an Enterprise Deployment."

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

This chapter contains the following sections:

- Section 9.1, "Overview of Extending the Domain for SOA Components"
- Section 9.2, "Prerequisites for Extending the Domain for Oracle SOA Components"
- Section 9.3, "Extending the Domain for SOA Components using the Configuration Wizard"
- Section 9.4, "Configuring Oracle Coherence for Deploying Composites"
- Section 9.5, "Post-Configuration and Verification Tasks"
- Section 9.6, "Configuring Oracle HTTP Server with the Extended Domain"
- Section 9.7, "Configuring a Default Persistence Store for Transaction Recovery"
- Section 9.8, "Configuring Oracle Adapters"
- Section 9.9, "Updating the B2B Configuration for High Availability"
- Section 9.10, "Backing Up the SOA Configuration"

9.1 Overview of Extending the Domain for SOA Components

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

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

Step	Description	More Information
Prepare for extending the Domain for SOA Components	Enable a virtual IP mapping for each of the hostnames, and synchronize the system clocks for the SOA WebLogic cluster	Section 9.2, "Prerequisites for Extending the Domain for Oracle SOA Components"
Extend the Domain for SOA Components	Extend the WebLogic domain you created in Chapter 8, "Creating a Domain for an Enterprise Deployment.".	Section 9.3, "Extending the Domain for SOA Components using the Configuration Wizard"
Configure Oracle Coherence for Deploying Composites	Configure Oracle Coherence in order to use unicast communication for deploying composites.	Section 9.4, "Configuring Oracle Coherence for Deploying Composites"
Post-Configuration and Verification Tasks	Follow these instructions for post-configuration and validation tasks.	Section 9.5, "Post-Configuration and Verification Tasks"
Configure the Oracle HTTP Server with the extended domain	Configure the Oracle HTTP Server with the managed servers, validate access, set the frontend HTTP host and port, and set the WLS Cluster address for the SOA_Cluster.	Section 9.6, "Configuring Oracle HTTP Server with the Extended Domain"
Configure a Default Persistence Store	Configure a default persistence store for transaction recovery.	Section 9.7, "Configuring a Default Persistence Store for Transaction Recovery"
Configure Oracle Adapters	Enable high availability for Oracle File and FTP Adapters, enable high availability for Oracle JMS Adapters, and scale the Oracle Database Adapter.	Section 9.8, "Configuring Oracle Adapters"
Update the B2B Configuration for High Availability	Set up File, FTP, or Email transports in a high availability environment.	Section 9.9, "Updating the B2B Configuration for High Availability"
Back Up the SOA Configuration	Back up the newly extended domain configuration.	Section 9.10, "Backing Up the SOA Configuration"

9.2 Prerequisites for Extending the Domain for Oracle SOA Components

Before you run the Configuration Wizard to extend the domain, enable a virtual IP mapping for each of the hostnames on the two SOA Machines, and synchronize the system clocks for the SOA WebLogic cluster.

This section includes the following topics:

- Section 9.2.1, "Enabling VIP2 on SOAHOST1 and VIP3 on SOAHOST2"

- Section 9.2.2, "Synchronize System Clocks"

9.2.1 Enabling VIP2 on SOAHOST1 and VIP3 on SOAHOST2

The SOA domain uses virtual hostnames as the listen addresses for the SOA managed servers. If you have not previously done so, you must enable a virtual IP mapping for each of these hostnames on the two SOA Machines, (VIP2 on SOAHOST1 and VIP3 on SOAHOST2), and correctly resolve the virtual hostnames in the network system used by the topology (either by DNS Server, hosts resolution).

To enable the virtual IPs, follow the steps described in Section 3.5, "Enabling Virtual IP Addresses for Administration Servers" if you have yet completed this procedure.

These virtual IPs and VHNs are required to enable server migration for the SOA Servers. You can configure server migration for the SOA system later for high availability purposes. Refer to Chapter 14, "Configuring Server Migration for an Enterprise Deployment" for more details on configuring server migration for the SOA servers.

9.2.2 Synchronize System Clocks

Oracle SOA uses Quartz to maintain its jobs and schedules in the database. Synchronize the system clocks for the SOA WebLogic cluster to enable proper functioning of jobs, adapters, and Oracle B2B.

9.3 Extending the Domain for SOA Components using the Configuration Wizard

Use the Configuration Wizard to extend the domain created in Chapter 8, "Creating a Domain for an Enterprise Deployment" to contain SOA components.

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

To extend the domain using the Configuration Wizard:

1. Change directory to the location of the Configuration Wizard. This is within the SOA home directory on SOAHOST1. Oracle recommends having all database instances up.

    ORACLE_COMMON_HOME/common/bin

2. Start the Configuration Wizard.

    ./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 following WebLogic domain directory

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

    Click **Next**.

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

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

    b. 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 the domain in Section 8.3, "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 Components Schema screen, do the following:

■ Select the **SOA Infrastructure**, **User Messaging Service**, and **SOA MDS Schema**.

■ For the Oracle RAC configuration for component schemas, select **Convert to GridLink**

Click **Next**.

8. The Configure Gridlink RAC Component Schema screen appears (Figure 9–1).

Figure 9–1 Configure GridLink RAC Component Schema Screen

In this screen enter values for the following fields, specifying the connect information for the Oracle RAC database that was seeded with RCU:

– **Driver:** Select **Oracle's driver (Thin) for GridLinkConnections,Versions:10 and later**.

– **Service Name:** Enter the service name of the database using lowercase characters. For example:

`soaedg.mycompany.com.`

– **Username:** Enter the database schema owner name of the corresponding component.

- **Password:** Enter the password for the database schema owner.

- Select **Enable FAN**

- Make sure **Enable SSL** is unchecked (alternatively if ssl is selected for ONS notifications to be encrypted, provide the appropriate wallet and wallet password).

- **Service listener**: Enter the SCAN address and port for the RAC database being used. You can identify this address by querying the appropriate parameter in the database using the TCP protocol:

```
SQL>show parameter remote_listener;

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

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

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

and

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

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

- **ONS Host**: Enter the SCAN address for the Oracle RAC database and the ONS remote port as reported by the database:

```
[orcl@db-scan1 ~]$ srvctl config nodeapps -s

ONS exists: Local port 6100, remote port 6200, EM port 2016
```

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

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

and

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

---

9. In the Test JDBC Data Sources screen, confirm that all connections were successful.

   The connections are tested automatically. The **Status** column displays the results. If all connections are not successful, 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
- JMS File Store

Click **Next**.

11. In the Select JMS Distributed Destination Type screen, ensure that ALL JMS system resources are configured to be **UDD**. This should be the default.

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

    a. Select the automatically created server and click **Rename** to change the name to **WLS_SOA1**.

    b. Click **Add** to add another new server and enter **WLS_SOA2** as the server name.

    c. Give servers **WLS_SOA1** and **WLS_SOA2** the attributes listed in Table 9–2. Do not modify the other servers that are shown in this screen; leave them as they are.

Table 9–2   Managed Servers

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled
WLS_SOA1	SOAHOST1VHN1	8001	n/a	No
WLS_SOA2	SOAHOST2VHN1	8001	n/a	No
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 9–3   Clusters

Name	Cluster Messaging Mode	Multicast Address	Multicast Port	Cluster Address
SOA_Cluster	unicast	n/a	n/a	SOAHOST1VHN1:8001,SOAHOST2VHN1:8001
WSM-PM_Cluster	unicast	n/a	n/a	Leave it empty.

Click **Next**.

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

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, delete the **LocalMachine** that appears by default and click the **Unix Machine** tab.

    The following entries appear (listed in Table 9–4):

    Table 9–4   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, select the SOA_Cluster and target the following to it:

- **usermessagingserver** and **usermessagingdriver-email**. (The usermessaging-xmpp, usermessaging-smpp, and usermessaging-voicexml applications are optional.)

- **oracle.sdp.***, **oracle.soa.*** and **oracle.rules.*** libraries

Select the **Admin Server** cluster and target the following to it:

oracle.rules.* library

Select the **WSM-PM_Cluster** and target the following to it:

wsm-pm application

For information on targeting applications and resources, see Appendix B, "Targeting Applications and Resources to Servers."

Click **Next**.

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

- Target **mds-owsm** to both **WSM-PM_Cluster** and **AdminServer**.

Click **Next**.

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

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

Click **Next**.

20. 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.

---

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

22. Restart the Administration Server using the procedure in Section 8.4.3, "Starting the Administration Server on SOAHOST1."

9.4 Configuring Oracle Coherence for Deploying Composites

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

Unicast communication does not enable nodes to discover other cluster members in this way. Consequently, you must specify the nodes that belong to the cluster. You do not need to specify all of the nodes of a cluster, however. You need only specify enough nodes so that a new node added to the cluster can discover one of the existing nodes. As a result, when a new node has joined the cluster, it is able to discover all of the other nodes in the cluster. Additionally, in configurations such as SOA enterprise

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

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

9.4.1 Enabling Communication for Deployment Using Unicast Communication

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 nine nodes as Well Known Addresses, but you can have more than nine nodes in the cluster. 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 9–4).

> **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.

> **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 9–2 Setting the Host Name Using the Start Server Tab of Oracle WebLogic Server Administration Console
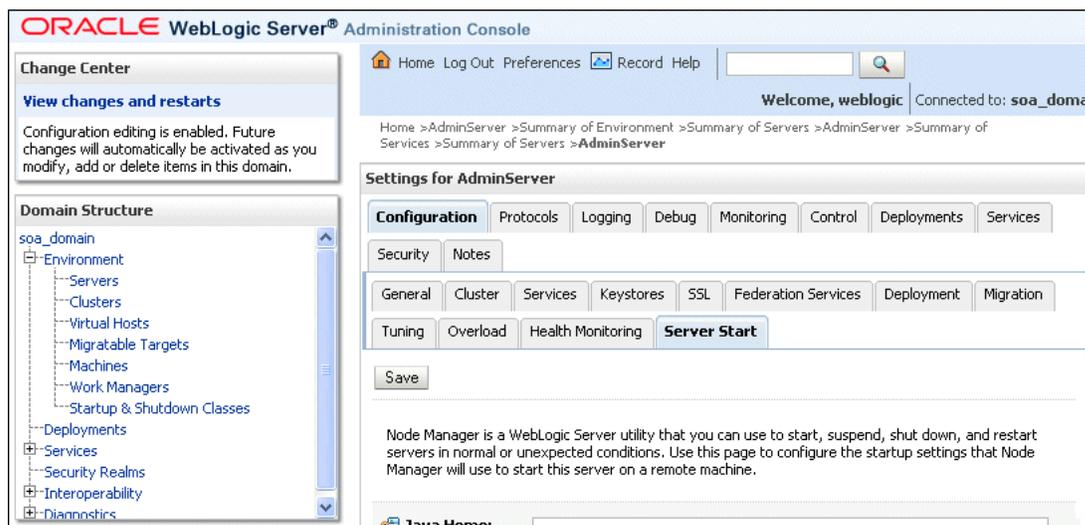
9.4.2 Specifying the Host Name Used by Oracle Coherence

Use the Administration Console to specify a host name used by Oracle Coherence.

To add the host name used by Oracle Coherence:

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 the **Name** column of the table. The settings page for the selected server appears.

5. Click **Lock & Edit**.

6. Click the **Server Start** tab (illustrated in Figure 9–2).

7. Enter the following for WLS_SOA1 and WLS_SOA2 into the Arguments field.

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

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*.

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
```

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.

9.5 Post-Configuration and Verification Tasks

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

This section includes the following topics:

- Section 9.5.1, "Disabling Host Name Verification for the WLS_SOAn Managed Servers"
- Section 9.5.2, "Restarting the Node Manager on SOAHOST1"
- Section 9.5.3, "Propagating the Domain Changes to the Managed Server Domain Directory"
- Section 9.5.4, "Starting and Validating the WLS_SOA1 Managed Server"
- Section 9.5.5, "Propagating the Domain Configuration to SOAHOST2 Using the unpack Utility"
- Section 9.5.6, "Starting and Validating the WLS_SOA2 Managed Server"
- Section 9.5.7, "Validating GridLink Data Sources"

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

For the enterprise deployment described in this guide, you set up the appropriate certificates to authenticate the different nodes with the Administration Server after you have completed the procedures to extend the domain for Oracle SOA Suite. You must disable the host name verification for the WLS_SOA1 and WLS_SOA2 managed servers to avoid errors when managing the different WebLogic Server instances. For more information, see Section 8.4.8, "Disabling Host Name Verification."

You enable host name verification again once the enterprise deployment topology configuration is complete. For more information, see Section 13.3, "Enabling Host Name Verification Certificates for Node Manager in SOAHOST1."

9.5.2 Restarting the Node Manager on SOAHOST1

Use the `startNodeManager.sh` script to restart Node Manager.

To restart the Node Manager on SOAHOST1:

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

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

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

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

   kill -9 9139
   ```

2. Start Node Manager:

   ```
   ./startNodeManager.sh
   ```

9.5.3 Propagating the Domain Changes to the Managed Server Domain Directory

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

To propagate start scripts and classpath configuration:

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

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

   ```
   cd ORACLE_COMMON_HOME/common/bin

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

   ```
   ./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.

   > **Note:** The configuration steps provided in this enterprise deployment topology are documented with the assumption that a local (per node) domain directory is used for each managed server.

9.5.4 Starting and Validating the WLS_SOA1 Managed Server

Before starting the WLS_SOA1 managed server please make sure the WLS__WSM1 managed server is up and running. Otherwise WLS_SOA1 will not start.

Start and validate the WLS_SOA1 managed server using the Administration Console.

To start the WLS_SOA1 managed server on SOAHOST1:

1. Start the WLS_SOA1 managed server using the Oracle WebLogic Server Administration Console as follows:

   a. Access the Administration Console at the following URL:

      ```
      http://ADMINVHN:7001/console
      ```

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

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

   c. Click **Servers**.

      The Summary of Servers screen appears.

   d. Click the **Control** tab.

   e. Select **WLS_SOA1** 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 Section 16.14, "Troubleshooting the Topology in an Enterprise Deployment" for possible causes.

3. Access the following URL to verify status of WLS_SOA1:

   ```
   http://SOAHOST1VHN1:8001/soa-infra/
   ```

   Access the following URL to verify the status of B2B:

   ```
   http://SOAHOST1VHN1:8001/b2bconsole/
   ```

   Access the following URL to verify status of the worklist application:

   ```
   http://SOAHOST1VHN1:8001/integration/worklistapp/
   ```

   Access the following URL to verify status of the composer application:

   ```
   http://SOAHOST1VHN1:8001/soa/composer/
   ```

   Before verifying access is granted, ensure that the WLS_WSM1 managed server is up and running.

9.5.5 Propagating the Domain Configuration to SOAHOST2 Using the unpack Utility

Propagate the domain you just configured to SOAHOST2 using the unpack utility.

To propagate the domain configuration:

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

   ```
   cd ORACLE_COMMON_HOME/common/bin

   scp soadomaintemplateExtSOA.jar oracle@SOAHOST2:ORACLE_COMMON_HOME/common/bin
   ```

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

```
cd ORACLE_COMMON_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.

Note: The configuration steps provided in this enterprise deployment topology are documented with the assumption that a local (per node) domain directory is used for each managed server.

9.5.5.1 Extracting the XEngine Files in SOAHOST2

To enable B2B's XEngine in SOAHOST2, extract the content of the XEngine tar file manually.

To extract the content of the XEngine tar file:

1. Run the following command from SOAHOST2:

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

2. Restart Node manager on SOAHOST2 using the procedure Section 9.5.2, "Restarting the Node Manager on SOAHOST1."

3. Verify the following directory structure after the extraction:

```
ORACLE_HOME/soa/soa/thirdparty/edifecs/
```

In this directory the following directories should appear:

- Common
- XEngine

In the XEngine directory the following directories and files should appear:

- bin
- config
- Readme.htm
- EULA.pdf

9.5.6 Starting and Validating the WLS_SOA2 Managed Server

Use the Administration Console to start the WLS_SOA2 managed server. Validate it by accessing soa-infra, b2bconsole, and worklistapp URLs.

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 Administration Console. If the server is shown as **Starting** or **Resuming**, wait for the server status to change to **Started**. If another status is reported (such as **Admin** or **Failed**), check the server output log files for errors. See Section 16.14, "Troubleshooting the Topology in an Enterprise Deployment" for possible causes.

3. Access the following URL for soa-infra:

   ```
   http://SOAHOST2VHN1:8001/soa-infra
   ```

4. Access the following URL to verify status of B2B:

   ```
   http://SOAHOST2VHN1:8001/b2bconsole
   ```

5. Access the following URL to verify status of the worklist application.

   ```
   http://SOAHOST2VHN1:8001/integration/worklistapp/
   ```

   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.

6. Access the following URL to verify status of the composer application.

   ```
   http://SOAHOST2VHN1:8001/soa/composer/
   ```

9.5.7 Validating GridLink Data Sources

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

To validate the GridLink data sources configuration:

1. Log on to the Oracle WebLogic Administration Console.

2. In the **Domain Structure** tree, expand **Services**, and select **Data Sources**.

3. Click one of the new data sources.

4. Click the **Monitoring** tab and select **Testing** subtab.

5. Select one of the servers and click **Test Data Source**.

6. Click the **Statistics** tab and select one of the servers.

7. Click the **ONS** tab, and then click the **Testing** tab.

8. Select the server and click **Test ONS**.

   If both tests are successful, the configuration is correct. If the ONS test fails, verify that the ONS service is running in the RAC database nodes:

   ```
   orcl@db-scan1 ~]$ srvctl status scan_listener
   SCAN Listener LISTENER_SCAN1 is enabled
   ```

```
SCAN listener LISTENER_SCAN1 is running on node db-scan1
SCAN Listener LISTENER_SCAN2 is enabled
SCAN listener LISTENER_SCAN2 is running on node db-scan2
SCAN Listener LISTENER_SCAN3 is enabled
SCAN listener LISTENER_SCAN3 is running on node db-scan2

[orcl@db-scan1 ~]$ srvctl config nodeapps -s
ONS exists: Local port 6100, remote port 6200, EM port 2016

[orcl@db-scan1 ~]$ srvctl status nodeapps | grep ONS
ONS is enabled
ONS daemon is running on node: db-scan1
ONS daemon is running on node: db-scan2
```

Run the ONS test from every WebLogic server that uses the data source.

9.6 Configuring Oracle HTTP Server with the Extended Domain

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

This section includes the following topics:

- Section 9.6.1, "Configuring Oracle HTTP Server for the WLS_SOAn Managed Servers"

- Section 9.6.2, "Validating Access Through Oracle HTTP Server"

- Section 9.6.3, "Setting the Frontend HTTP Host and Port"

9.6.1 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, set the WebLogicCluster parameter to the list of nodes in the cluster.

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

To enable Oracle HTTP Server to route to the SOA_Cluster:

1. On WEBHOST1 and WEBHOST2, add directives to the soa.vh.conf file located in the following directory:

    ORACLE_BASE/admin/*instance_name*/config/OHS/*component_name*/moduleconf

    Note that this assumes you created the soa_vh.conf file using the instructions in Section 7.6, "Defining Virtual Hosts."

    Add the following directives inside the <VirtualHost> tags:

    ```
    <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>

<Location /b2b/services/ws/>
        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>

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

The `soa_vh.conf` file will appear as it does in Example 9–1.

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
```

Example 9–1   soa_vh.conf file

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

<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>

<Location /b2b/services/ws/>
    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>

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

> **Note:** If internal invocations are going to be used in the system, add the appropriate locations to the soainternal virtual host.

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. 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.

9.6.2 Validating Access Through Oracle HTTP Server

Verify that the server status is reported as **Running** in the Administration Console. If the server is shown as **Starting** or **Resuming**, wait for the server status to change to **Started**. If another status is reported (such as **Admin** or **Failed**), check the server output log files for errors. See Section 16.14, "Troubleshooting the Topology in an Enterprise Deployment" for possible causes.

Verify that you can access these URLs, where 'webhostN' specifies the name of each Oracle HTTP Server host. Check these URLS for both WEBHOST1 and 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.

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

9.6.3 Setting the Frontend HTTP Host and Port

This section contains the procedure for setting the frontend HTTP host and port using the Administration Console. It also contains information about how callback URLs are calculated.

9.6.3.1 Setting the Frontend HTTP Host and Port

To 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

   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.
   ```

6. Click **Save**.

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

8. Restart the servers for the Frontend Host directive in the cluster to take affect.

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 functionality 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).

9.6.3.2 About the Callback URL

This section describes how the SOA system calculates the callback URL.

- If a request to SOA originates from an external or internal service, SOA uses the callback URL specified by the client.

- If a request to an external or internal asynchronous service originates from SOA, SOA uses the following method, in decreasing order of preference to calculate the callback URL:

    - 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.

    - 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)

    - 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*.

    - Use the local host name as provided by WLS MBean APIs. Oracle does not recommend this in high availability environments such as enterprise deployment.

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

To set the location for the default persistence stores:

1. Log into the Oracle WebLogic Server Administration Console.

2. In the Change Center section, click **Lock & Edit**.

3. In the Domain Structure window, expand the **Environment** node and then click the **Servers** node. The Summary of Servers page appears.

4. 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.

5. Click the **Configuration** tab, and then the **Services** tab.

6. In the Default Store section of the page, enter the path to the folder where the default persistent stores will store its data files. The directory structure of the path is as follows:

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

7. Repeat steps 3, 4, 5, and 6 for the WLS_SOA2 server.

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

9. Restart both SOA servers.

10. Verify that the following files are created in the following directory after WLS_SOA1 and WLS_SOA2 are restarted:

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

    - _WLS_WLS_SOA1000000.DAT
    - _WLS_WLS_SOA2000000.DAT

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

9.8 Configuring Oracle Adapters

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

This section includes the following topics:

- Section 9.8.1, "Enabling High Availability for Oracle File and FTP Adapters"
- Section 9.8.2, "Enabling High Availability for Oracle JMS Adapters"
- Section 9.8.3, "Scaling the Oracle Database Adapter"

9.8.1 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 the *Oracle Fusion Middleware User's Guide for Technology Adapters*. The database mutex locking operation enables these adapters to ensure that multiple references do not overwrite one another if they write to the same directory.

Note: The operations described in this section are 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.

9.8.1.1 Using the Database Mutex Locking Operation

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:** If you use database as a coordinator, increase global transaction timeouts.

To make outbound Oracle File or FTP Adapters highly available, modify Oracle File Adapter deployment descriptor for the connection-instance corresponding to `eis/HAFileAdapter` from the Oracle WebLogic Server console:

1. Log into your Oracle WebLogic Server console. To access the console navigate to the following URL:

   ```
   http://servername:portnumber/console
   ```

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

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

4. Click the **Configuration** tab.

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

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

7. The connection factory properties appear as shown in Figure 9–3.

Figure 9–3   Oracle WebLogic Server Console - Settings for javax.resource.cci.Connectionfactory Page
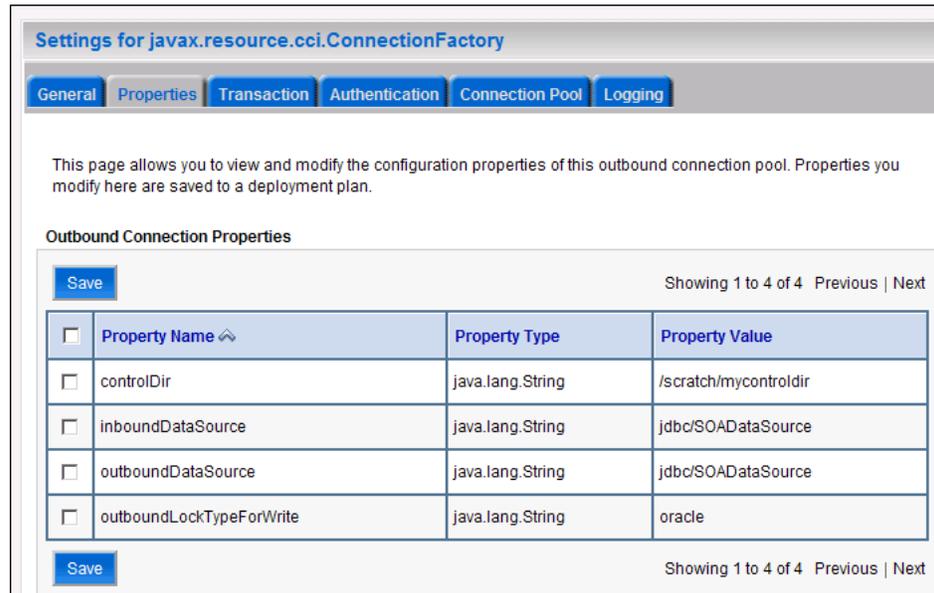
Click on **Lock & Edit**. After this, the property value column becomes editable (you can click on any of the rows under "Property Value" and modify its value).

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

Table 9–5   Oracle File and FTP Adapters

Parameter	Description
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: `ASERVER_HOME/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 in the following directory: `ORACLE_HOME/rcu/integration/soainfra/sql/createschema_soainfra_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.

Table 9–5 (Cont.) Oracle File and FTP Adapters

Parameter	Description

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 are located in the following directory:
	
	`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 data source 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.

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

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

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

10. Click **Save and Activate**.

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

    In the Administration Console, click **Deployments** in the left pane for **Domain Structure**.

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

12. Configure BPEL Process or Mediator Scenario to use the connection factory as shown in the following example (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.

> **Note:** Perform the same steps for updating the control dir for the FTPAdapter. Use the `eis/Ftp/HAFtpAdapter` connection factory instance for these modifications.

9.8.2 Enabling High Availability for Oracle JMS Adapters

When the Oracle JMS adapter communicates with multiple servers in a cluster, the adapter's connection factory property `FactoryProperties` must list available servers. If it does not list servers, the connection establishes to only one random server. If that particular server goes down, no further messages are processed.

To verify that the adapter's JCA connection factory:

1. Log into your Oracle WebLogic Server Administration Console using the following URL:

   `http://servername:portnumber/console`

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

3. Click **JMSAdapter** under Summary of Deployments on the right pane.

4. Click the **Configuration** tab.

5. Click the Outbound Connection Pools tab and expand `oracle.tip.adapter.jms.IJmsConnectionFactory` to see the configured connection factories.

6. Click the specific instance you are using (for example, `eis/wls/Queue`). The Outbound Connection Properties for the connection factory opens.

7. Click **Lock & Edit**.

8. In the FactoryProperties field (click on the corresponding cell under Property value), enter the following:

   ```
   java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory;
   java.naming.provider.url=t3://soahostvhn1:8001,soahost2vhn1:8001;java.naming.se
   curity.principal=weblogic;java.naming.security.credentials=mypassword
   ```

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

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

    ASERVER_HOME/*cluster_name*/dp/Plan.xml

11. Click **Save and Activate**.

Update the deployment in the console:

1. Click **Deployments** and select the JMS Adapter.

2. Click **Lock & Edit** then **Update**.

3. Select **Update this application in place with new deployment plan changes (A deployment plan must be specified for this option.)** and select the deployment plan saved in a shared storage location; all servers in the cluster must be able to access the plan).

4. Click **Finish**.

5. Activate the changes.

9.8.3 Scaling the Oracle Database Adapter

The introduction of skip locking has superseded the previous best practice of using `LogicalDeletePollingStrategy` or `DeletePollingStrategy` with a unique `MarkReservedValue` on each polling node, and setting `MaxTransactionSize`. 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}`.

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 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.

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

9.9 Updating the B2B Configuration for High Availability

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:

1. Log in to Oracle Enterprise Manager Fusion Middleware Control with the username and password specified in Section 8.4.1, "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#**.

    Enter this value in only one of the two servers.

9. Click **Invoke**.

9.10 Backing Up the SOA Configuration

After you have verified that the extended domain is working, back up the domain configuration. This is a quick backup for the express purpose of immediate restore in case of failures in future procedures. Back up the configuration to the local disk. This backup can be discarded once you have completed the enterprise deployment. Once you have completed the enterprise deployment, you can initiate the regular deployment-specific backup and recovery process.

For information about backing up the environment, see "Backing Up Your Environment" in the *Oracle Fusion Middleware Administrator's Guide*. For information about recovering your information, see "Recovering Your Environment" in the *Oracle Fusion Middleware Administrator's Guide*.

To back up the domain configuration:

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 are located in the following directory:

```
ORACLE_BASE/ admin/domain_name
```

To back up the Administration Server run the following command on SOAHOST1:

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

> **Note:** Back up *ORACLE_HOME* if any changes are made to the XEngine configuration that is part of your B2B setup. These files are located in the following directory:
>
> ```
> ORACLE_HOME/soa/thirdparty/edifecs/XEngine
> ```
> To back up *ORACLE_HOME*:
>
> ```
> tar -cvpf fmwhomeback.tar MW_HOME
> ```

10

Extending the Domain to Include Oracle BPM

This chapter describes the procedures for extending the domain to include Oracle BPM.

This chapter contains the following section:

- Section 10.1, "Overview of Extending the Domain to include Oracle BPM"
- Section 10.2, "Option 1: Extending a Domain to Include SOA and BPM"
- Section 10.3, "Option 2: Extending a SOA Domain to Include Oracle BPM"
- Section 10.4, "Backing Up the Oracle BPM Configuration"

10.1 Overview of Extending the Domain to include Oracle BPM

You can install and configure Oracle 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). Table 10–1 summarizes this approach. For configuration steps, see Section 10.2, "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. Table 10–2 summarizes this approach. For configuration steps, see Section 10.3, "Option 2: Extending a SOA Domain to Include Oracle BPM."

Table 10–1   Steps for Extending tan existing Domain to include SOA and BPM

Step	Description	More Information
Enabling VIP2 on SOAHOST1 and VIP3 on SOAHOST2	Enable a virtual IP mapping for each of the hostnames.	Section 10.2.1, "Enabling VIP2 on SOAHOST1 and VIP3 on SOAHOST2"
Run the Configuration Wizard to Extend the 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.	Section 10.2.2, "Running the Configuration Wizard on SOAHOST1 to Extend the Current Domain"
Configure Oracle Coherence for Deploying Composites	Configure Oracle Coherence in order to use unicast communication for deploying composites.	Section 10.2.3, "Configuring Oracle Coherence for Deploying Composites"

Table 10–1   (Cont.) Steps for Extending tan existing Domain to include SOA and BPM

Step	Description	More Information
Disabling Host Name Verification for the WLS_SOAn Managed Servers	Set up the appropriate certificates to authenticate the different nodes with the Administration Server after you have completed the procedures to extend the domain for Oracle BPM.	Section 10.2.4, "Disabling Host Name Verification for the WLS_SOAn Managed Servers"
Propagate the Domain Changes to the Managed Server Domain Directory	Propagate the start scripts and classpath configuration from the Administration Server's domain directory to the managed server domain directory.	Section 10.2.5, "Propagating the Domain Changes to the Managed Server Domain Directory"
Start and Validate the WLS_SOA1 Managed Server	Start and validate the WLS_SOA1 managed server using the Oracle WebLogic Server Administration Console.	Section 10.2.6, "Starting and Validating the WLS_SOA1 Managed Server"
Propagate the Domain Configuration to SOAHOST2	Propagate the Domain Configuration to SOAHOST2 Using the Unpack Utility.	Section 10.2.7, "Propagating the Domain Configuration to SOAHOST2 Using the Unpack Utility"
Extract the XEngine Files in SOAHOST2	To enable B2B's XEngine in SOAHOST2, extract the content of the XEngine tar manually.	Section 10.2.8, "Extracting the XEngine Files in SOAHOST2"
Start and Validate the WLS_SOA2 Managed Server	Start the WLS_SOA2 managed server and check that it is configured correctly.	Section 10.2.9, "Starting and Validating the WLS_SOA2 Managed Server"
Configure the Oracle HTTP Server for WLS_SOAn Managed Servers	Enable Oracle HTTP Server to route to the SOA_Cluster.	Section 10.2.11, "Configuring Oracle HTTP Server for WLS_SOAn Managed Servers"
Validating Access Through Oracle HTTP Server	Verify that the server status is reported as Running.	Section 10.2.12, "Validating Access Through Oracle HTTP Server"
Setting the Frontend HTTP Host and Port	Set the frontend HTTP host and port for the Oracle WebLogic Server cluster.	Section 10.2.13, "Setting the Frontend HTTP Host and Port"
Configure a Default Persistence Store for Transaction Recovery	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.	Section 10.2.14, "Configuring a Default Persistence Store for Transaction Recovery"
Enable High Availability for Oracle File and FTP Adapters	Make Oracle File and FTP Adapters highly available for outbound operations using the database mutex locking operation.	Section 10.2.15, "Enabling High Availability for Oracle File and FTP Adapters"

Table 10–2   Steps for Extending an Existing Domain that Already includes SOA

Step	Description	More Information
Run 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.	Section 10.3.1, "Running the Configuration Wizard on SOAHOST1 to Extend a SOA Domain to Include BPM"
Propagate 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.	Section 10.3.2, "Propagating the Domain Configuration to the managed server directory in SOAHOST1 and to SOAHOST2"
Start the BPM Suite Components	For configuration changes and start scripts to be effective, restart the WLS_SOAn server to which BPM has been added.	Section 10.3.3, "Starting the BPM Suite Components"
Configure Oracle HTTP Server for the WLS_SOAn Managed Servers	Enable Oracle HTTP Server to route to BPM web applications by setting the WebLogicCluster parameter to the list of nodes in the cluster.	Section 10.3.4, "Configuring Oracle HTTP Server for the WLS_SOAn Managed Servers"
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.	Section 10.3.5, "Validating Access Through Oracle HTTP Server"

Prerequisites for Extending the Domain to Include Oracle BPM

Before extending the current domain, ensure that your existing deployment meets the following prerequisites:

- **Back up the installation** - If you have not yet backed up the existing Fusion Middleware Home and domain, Oracle recommends backing it up now.

  To back up the existing Fusion Middleware Home and domain run the following command on SOAHOST1:

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

- There is an existing WL_HOME and ORACLE_HOME installed in previous chapters on a shared storage.

10.2 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 BPM. 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.

> **Note:** 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 10.2.1, "Enabling VIP2 on SOAHOST1 and VIP3 on SOAHOST2"
- Section 10.2.2, "Running the Configuration Wizard on SOAHOST1 to Extend the Current Domain"
- Section 10.2.3, "Configuring Oracle Coherence for Deploying Composites"
- Section 10.2.4, "Disabling Host Name Verification for the WLS_SOAn Managed Servers"
- Section 10.2.5, "Propagating the Domain Changes to the Managed Server Domain Directory"
- Section 10.2.6, "Starting and Validating the WLS_SOA1 Managed Server"
- Section 10.2.7, "Propagating the Domain Configuration to SOAHOST2 Using the Unpack Utility"
- Section 10.2.8, "Extracting the XEngine Files in SOAHOST2"
- Section 10.2.9, "Starting and Validating the WLS_SOA2 Managed Server"
- Section 10.2.10, "Validating GridLink Data Sources"
- Section 10.2.11, "Configuring Oracle HTTP Server for WLS_SOAn Managed Servers"
- Section 10.2.12, "Validating Access Through Oracle HTTP Server"
- Section 10.2.13, "Setting the Frontend HTTP Host and Port"
- Section 10.2.14, "Configuring a Default Persistence Store for Transaction Recovery"
- Section 10.2.15, "Enabling High Availability for Oracle File and FTP Adapters"
- Section 10.2.16, "Enabling High Availability for Oracle JMS Adapters"
- Section 10.2.17, "Scaling the Oracle Database Adapter"
- Section 10.2.18, "Updating the B2B Configuration for High Availability"

10.2.1 Enabling VIP2 on SOAHOST1 and VIP3 on SOAHOST2

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 virtual IPs (VIP2 and VIP3). These procedures are required for server migration.

To enable the virtual IP, see Section 3.5, "Enabling Virtual IP Addresses for Administration Servers."

10.2.2 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.

To extend the domain for Oracle BPM:

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.

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

3. Start the Oracle Fusion Middleware Configuration Wizard:

   ```
   ./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 Components Schema screen, do the following:

   - Select the **SOA Infrastructure**, **User Messaging Service**, and **SOA MDS Schema**.

   - For the Oracle RAC configuration for component schemas, select **Convert to GridLink**.

   Click **Next**.

8. The Configure Gridlink RAC Component Schema screen appears (Figure 10–1).

Figure 10–1   Configure GridLink RAC Component Schema Screen

In this screen enter values for the following fields, specifying the connect information for the Oracle RAC database that was seeded with RCU:

– **Driver:** Select **Oracle's driver (Thin) for GridLinkConnections,Versions:10 and later**.

– **Service Name:** Enter the service name of the database using lowercase characters. For example:

```
soaedg.mycompany.com
```

– **Username:** Enter the database schema owner name of the corresponding component. The user names shown in (replace table reference) assume that `soedg` was used as prefix for schema creation from RCU.

– **Password:** Enter the password for the database schema owner.

– Select **Enable FAN**

– Make sure **Enable SSL** is unchecked (alternatively if ssl is selected for ONS notifications to be encrypted, provide the appropriate wallet and wallet password).

– **Service listener**: Enter the SCAN address and port for the RAC database being used. You can identify this address by querying the appropriate parameter in the database using the TCP protocol:

```
SQL>show parameter remote_listener;

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

> **Note:** For Oracle Database 11*g* Release 1 (11.1), use the virtual IP and port of each database instance listener, for example:
>
> ```
> custdbhost1-vip.mycompany.com (port 1521)
> ```
>
> and
>
> ```
> custdbhost2-vip.mycompany.com (1521)
> ```
>
> For Oracle Database 10*g*, use multi data sources to connect to an Oracle RAC database. For information about configuring multi data sources see Appendix A, "Using Multi Data Sources with Oracle RAC."

- **ONS Host**: Enter the SCAN address for the Oracle RAC database and the ONS remote port as reported by the database:

  ```
  [orcl@db-scan1 ~]$ srvctl config nodeapps -s
  
  ONS exists: Local port 6100, remote port 6200, EM port 2016
  ```

  > **Note:** For Oracle Database 11*g* Release 1 (11.1), use the hostname and port of each database's ONS service, for example
  >
  > ```
  > custdbhost1.mycompany.com (port 6200)
  > ```
  >
  > and
  >
  > ```
  > custdbhost2.mycompany.com (6200)
  > ```

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
    - JMS File Stores
    - Managed Servers, Clusters, and Machines
    - Deployments and Services

    Click **Next**.

11. Once the new deployment plan has been saved and activated, activate the FileAdapter deployment (the deployment remains in Prepared state if not started).

    To activate the FileAdapter deployment plan:

    a. In the Administration Console, click **Deployments** in the left pane for **Domain Structure**.

    b. Select the **FileAdapter** under **Summary of Deployments** on the right pane.

    c. Select **Start**, and then **Servicing All Requests**.

12. 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

13. 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 10–3. Do not modify the other servers that appear in this screen; leave them as they are.

Table 10–3   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**.

14. In the Configure Clusters screen, add the cluster listed in Table 10–4. Do not modify the other clusters that display in this screen; leave them as they are.

Table 10–4   Cluster

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

Click **Next**.

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

- **SOA_Cluster:**
  - WLS_SOA1
  - WLS_SOA2

Click **Next**.

16. 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 10–5):

Table 10–5   Machines

Name	Node Manager Listen Address
SOAHOST1	SOAHOST1
SOAHOST2	SOAHOST2

Table 10–5   (Cont.) Machines

Name	Node Manager Listen Address
ADMINHOST	localhost

Leave all other fields to their default values.

Click **Next**.

17. In the Assign Servers to Machines screen, assign the servers to machines according to the following table:

Table 10–6   Assigning Servers to Machines

Server	Machine
AdminServer	ADMINHOST
WLS_SOA1	SOAHOST1
WLS_SOA2	SOAHOST2
WLS_WSM1	SOAHOST1
WLS_WSM2	SOAHOST2

Click **Next**.

18. 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**.

    ■ Target the **oracle.rules.*** library only to **Admin Server** and **SOA_Cluster**.

    Click **Next**.

19. In the Target Services to Clusters or Servers screen, target the **mds-owsm**, **mdw-owsm-rac0** and **mds-owsm-rac1** datasources to the **WSM-PM_Cluster** and the **AdminServer** and Click **Next**.

    For information on targeting applications and resources, see Appendix B, "Targeting Applications and Resources to Servers."

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

    ASERVER_HOME/soa_cluster_name/jms

    Select **Direct-write** policy for all stores.

    Click **Next**.

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

22. 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 8.4.3, "Starting the Administration Server on SOAHOST1."

10.2.3 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.

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 system, you must configure Oracle Coherence to use a specific host name to create the Oracle Coherence cluster.

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

10.2.3.1 Enabling Communication for Deployment Using Unicast Communication

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 10–2).

> **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.

> **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 10–2   Setting the Host Name Using the Start Server Tab of Oracle WebLogic Server Administration Console
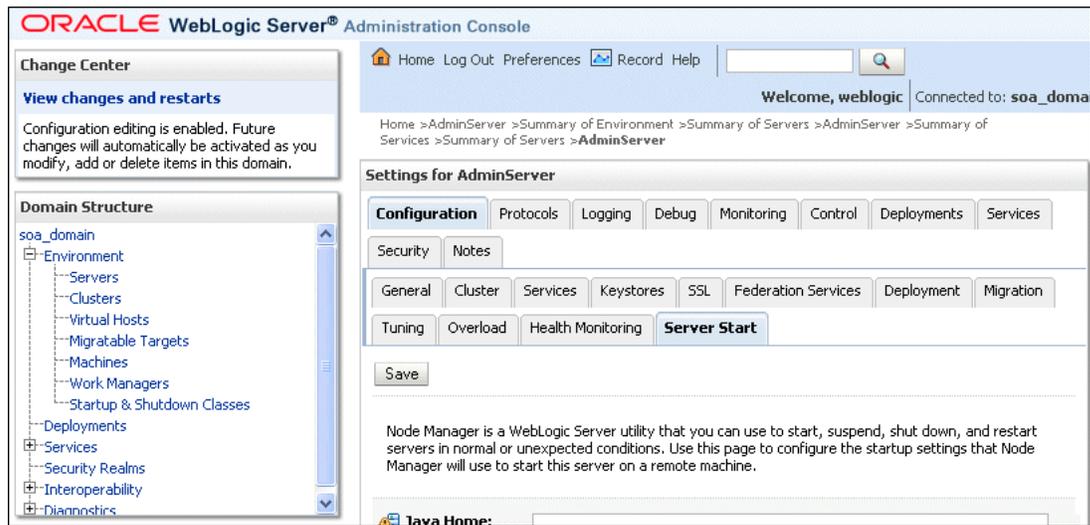
10.2.3.2 Specifying the Host Name Used by Oracle Coherence

Use the Administration Console to specify a host name used by Oracle Coherence. This procedure is required for both servers WLS_SOA1 and WLS_SOA2.

To add the host name used by Oracle Coherence:

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, represented as hyperlinks, in the **Name** column of the table.

    The settings page for the selected server appears.

5. Click **Lock & Edit**.

6. Click the **Server Start** tab (illustrated in Figure 10–2).

7. Enter the following for WLS_SOA1 and WLS_SOA2 into the Arguments field.

    > **Note:**   There should be no breaks in lines between the different -D parameters. Do not copy or paste the text to your Administration Console's arguments text field. It 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.

> **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*.

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
```

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.

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

For the enterprise deployment described in this guide, you set up the appropriate certificates to authenticate the different nodes with the Administration Server after you have completed the procedures to extend the domain for Oracle SOA Suite. You must disable the host name verification for the WLS_SOA1 and WLS_SOA2 managed servers to avoid errors when managing the different WebLogic Server instances. For more information, see Section 8.4.8, "Disabling Host Name Verification."

You enable host name verification again once the enterprise deployment topology configuration is complete. For more information, see Section 13.3, "Enabling Host Name Verification Certificates for Node Manager in SOAHOST1."

10.2.5 Propagating the Domain Changes to the Managed Server Domain Directory

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

Note: If you configure server migration before unpacking (extending the domain to include BPM), the `wlsifconfig` script is overwritten and must be restored from the copy that the `-overwrite` option provides.

To propagate start scripts and classpath configuration:

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

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

    ```
    cd ORACLE_COMMON_HOME/common/bin

    ./pack.sh -managed=true -domain=ASERVER_HOME
     -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:

    ```
    ./unpack.sh -domain=MSERVER_HOME
    -overwrite_domain=true
    -template=soadomaintemplateExtSOABPM.jar
    -app_dir=MSERVER_HOME/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.

Note: The configuration steps provided in this enterprise deployment topology are documented with the assumption that a local (per node) domain directory is used for each managed server.

10.2.6 Starting and Validating the WLS_SOA1 Managed Server

Start and validate the WLS_SOA1 managed server using the Oracle WebLogic Server Administration Console.

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 16.14, "Troubleshooting the Topology in an Enterprise Deployment" 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.

10.2.7 Propagating the Domain Configuration to SOAHOST2 Using the Unpack Utility

Propagate the domain configuration to SOAHOST2 using the unpack utility.

> **Note:** If you configure server migration before unpacking (extending the domain to include BPM), the `wlsifconfig` script is overwritten and must be restored from the copy that the `-overwrite` option provides.

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.

   cd *ORACLE_COMMON_HOME*/common/bin

   scp soadomaintemplateExtSOABPM.jar oracle@node2:*ORACLE_COMMON_HOME*/common/bin

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

```
cd ORACLE_COMMON_HOME/common/bin

./unpack.sh -domain=MSERVER_HOME
-overwrite_domain=true -template=soadomaintemplateExtSOABPM.jar
-app_dir=MSERVER_HOME/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.

> **Note:** The configuration steps provided in this enterprise deployment topology are documented with the assumption that a local (per node) domain directory is used for each managed server.

10.2.8 Extracting the XEngine Files in SOAHOST2

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

1. Run the following command from SOAHOST2:

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

2. Restart Node manager on SOAHOST2 using the procedure Section 9.5.2, "Restarting the Node Manager on SOAHOST1."

3. Verify the following directory structure after the extraction:

```
ORACLE_HOME/soa/soa/thirdparty/edifecs/
```

In this directory the following directories should appear:

- Common
- XEngine

In the XEngine directory the following directories and files should appear:

- bin
- config
- Readme.htm
- EULA.pdf

10.2.9 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 16.14, "Troubleshooting the Topology in an Enterprise Deployment" 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.

10.2.10 Validating GridLink Data Sources

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

To validate the GridLink data sources configuration:

1. Log on to the Oracle WebLogic Administration Console.

2. In the **Domain Structure** tree, expand **Services**, and select **Data Sources**.

3. Click one of the new data sources.

4. Click the **Monitoring** tab and select one of the servers.

5. Click the **Statistics** tab and select one of the servers.

6. Click the **ONS** tab, and then click the **Testing** tab.

7. Select the server and click **Test ONS**.

If both tests are successful, the configuration is correct. If the ONS test fails, verify that the ONS service is running in the RAC database nodes:

```
orcl@db-scan1 ~]$ srvctl status scan_listener
SCAN Listener LISTENER_SCAN1 is enabled
SCAN listener LISTENER_SCAN1 is running on node db-scan1
SCAN Listener LISTENER_SCAN2 is enabled
SCAN listener LISTENER_SCAN2 is running on node db-scan2
SCAN Listener LISTENER_SCAN3 is enabled
SCAN listener LISTENER_SCAN3 is running on node db-scan2

[orcl@db-scan1 ~]$ srvctl config nodeapps -s
ONS exists: Local port 6100, remote port 6200, EM port 2016

[orcl@db-scan1 ~]$ srvctl status nodeapps | grep ONS
ONS is enabled
ONS daemon is running on node: db-scan1
ONS daemon is running on node: db-scan2
```

Run the ONS test from every WebLogic server that uses the data source.

10.2.11 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, set the `WebLogicCluster` parameter to the list of nodes in the cluster.

To enable Oracle HTTP Server to route to the SOA_Cluster:

1. On WEBHOST1 and WEBHOST2, add directives to the `soa.vh.conf` file located in the following directory:

    ORACLE_INSTANCE/config/OHS/*component_name*/moduleconf

    Note that this assumes you created the `soa_vh.conf` file using the instructions in Section 7.6, "Defining Virtual Hosts."

    Add the following directives inside the `<VirtualHost>` tags:

    ```
    # 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>
    ```

    The `soa_vh.conf` file will appear as it does in Example 10–1.

2. Restart Oracle HTTP Server on WEBHOST1 and WEBHOST2:

    ```
    WEBHOST1> ORACLE_INSTANCE/bin/opmnctl restartproc ias-component=ohs1
    WEBHOST2> ORACLE_INSTANCE/bin/opmnctl restartproc ias-component=ohs2
    ```

Example 10–1   soa_vh.conf file

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

<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>

# 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>
</VirtualHost>
```

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 is 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.

10.2.12 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 16.14, "Troubleshooting the Topology in an Enterprise Deployment" 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 3.3, "Configuring the Load Balancers."

10.2.13 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

   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.
   ```

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 functionality 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:

  - 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.

  - 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).

  - 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*.

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

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

To set the location for the default persistence stores:

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** sub-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:

   ASERVER_HOME/*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.

10.2.15 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 operations described in this section are 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 Oracle RAC backend or in the SOA managed servers.

10.2.15.1 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 10–3.

Figure 10–3   Oracle WebLogic Server Console - Settings for javax.resource.cci.Connectionfactory Page

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

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

Table 10–7 Oracle File and FTP Adapters

Parameter	Description
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: *ASERVER_HOME*/*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 in the following directory: *ORACLE_HOME*/rcu/integration/soainfra/sql/createschema_ soainfra_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 are located in the following directory: *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 data source 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:

```
ASERVER_HOME/cluster_name/dd/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.

10.2.16 Enabling High Availability for Oracle JMS Adapters

When the Oracle JMS adapter communicates with multiple servers in a cluster, the adapter's connection factory property `FactoryProperties` must list available servers. If it does not list servers, the connection establishes to only one random server. If that particular server goes down, no further messages are processed.

To verify that the adapter's JCA connection factory:

1. Log into your Oracle WebLogic Server Administration Console using the following URL:

   `http://servername:portnumber/console`

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

3. Click **JMSAdapter** under Summary of Deployments on the right pane.

4. Click the **Configuration** tab.

5. Click the Outbound Connection Pools tab and expand `oracle.tip.adapter.jms.IJmsConnectionFactory` to see the configured connection factories.

6. Click the specific instance you are using (for example, `eis/wls/Queue`). The Outbound Connection Properties for the connection factory opens.

7. Click **Lock & Edit**.

8. In the FactoryProperties field (click on the corresponding cell under Property value), enter the following:

   ```
   java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory;
   java.naming.provider.url=t3://soahostvhn1:8001,soahost2vhn1:8001;java.naming.se
   curity.principal=weblogic;java.naming.security.credentials=mypassword
   ```

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

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

ASERVER_HOME/*cluster_name*/dp/Plan.xml

11. Click **Save and Activate**.

Update the deployment in the console:

1. Click **Deployments** and select the JMS Adapter.

2. Click **Lock & Edit** then **Update**.

3. Select **Update this application in place with new deployment plan changes (A deployment plan must be specified for this option.)** and select the deployment plan saved in a shared storage location; all servers in the cluster must be able to access the plan).

4. Click **Finish**.

5. Activate the changes.

10.2.17 Scaling the Oracle Database Adapter

The introduction of skip locking has superseded the previous best practice of using `LogicalDeletePollingStrategy` or `DeletePollingStrategy` with a unique `MarkReservedValue` on each polling node, and setting `MaxTransactionSize`. 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}`.

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`.

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

10.2.18 Updating the B2B Configuration for High Availability

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:

1. Log in to Oracle Enterprise Manager Fusion Middleware Control with the username and password specified in

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#**.

   Enter this value in only one of the two servers.

9. Click **Invoke**.

10.3 Option 2: Extending a SOA Domain to Include Oracle BPM

In this step, you extend the domain created in Section 9, "Extending the Domain for SOA Components" to include Oracle BPM.

Prerequisites for Extending the SOA Domain to Include Oracle BPM

Before extending the current domain, ensure that your existing deployment meets the following prerequisites:

- **Back up the installation** - If you have not yet backed up the existing Fusion Middleware Home and domain, Oracle recommends backing it up now.

  To back up the existing Fusion Middleware Home and domain:

  ```
  tar -cvpf fmwhomeback.tar MW_HOME
  tar -cvpf domainhomeback.tar ASERVER_HOME
  ```

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

- There is an existing WL_HOME and SOA ORACLE_HOME (binaries) are installed in previous chapters on a shared storage and are available from SOAHOST1 and SOAHOST2 (this is required before the WebLogic Configuration Wizard steps are performed to extend the domain).

- 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 10.3.1, "Running the Configuration Wizard on SOAHOST1 to Extend a SOA Domain to Include BPM"

- Section 10.3.2, "Propagating the Domain Configuration to the managed server directory in SOAHOST1 and to SOAHOST2"

- Section 10.3.3, "Starting the BPM Suite Components"

-

-

10.3.1 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.

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

2. Start the Oracle Fusion Middleware Configuration Wizard:

   ```
   ./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**.

   Oracle BPM uses the same Data Sources 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.sdp.***, **oracle.bpm.***, and **oracle.soa.*** libraries only to **SOA_Cluster**.

   - Target the **oracle.rules.*** library to **SOA_Cluster** and **Admin Server**.

Click **Next**.

For information on targeting applications and resources, see Appendix B, "Targeting Applications and Resources to Servers."

10. In the Target Services to Clusters or Servers screen, target the **mds-owsm** datasource to the **WSM-PM_Cluster** and the **AdminServer** and click **Next**.

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

    ASERVER_HOME/*soa_cluster_name*/jms

    Select **Direct-write** policy for all stores.

    Click **Next**.

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

13. 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 8.4.3, "Starting the Administration Server on SOAHOST1."

10.3.2 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.

> **Note:** If you configure server migration before unpacking (extending the domain to include BPM), the `wlsifconfig` script is overwritten and must be restored from the copy that `-overwrite` option provides.

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:

   ```
   cd ORACLE_COMMON_HOME/common/bin

   ./pack.sh -managed=true -domain=ASERVER_HOME
    -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:

   ```
   ./unpack.sh -domain=MSERVER_HOME
   -overwrite_domain=true -template=soadomaintemplateExtSOABPM.jar
   -app_dir=MSERVER_HOME/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.

4. Copy the template to SOAHOST2:

   cd *ORACLE_COMMON_HOME*/common/bin

   scp soadomaintemplateExtSOABPM.jar oracle@SOAHOST2:/*ORACLE_HOME*/common/bin

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

   cd *ORACLE_COMMON_HOME*/common/bin

   ./unpack.sh -domain=*MSERVER_HOME*/*DOMAIN_NAME* -overwrite_domain=true
   -template=soadomaintemplateExtSOABPM.jar -app_dir=*MSERVER_HOME*/applications

> **Note:** The configuration steps provided in this enterprise deployment topology are documented with the assumption that a local (per node) domain directory is used for each managed server.

10.3.3 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. 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**.

10.3.4 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, set the `WebLogicCluster` parameter to the list of nodes in the cluster.

To enable Oracle HTTP Server to route to the SOA_Cluster:

1. On WEBHOST1 and WEBHOST2, add directives to the `soa.vh.conf` file located in the following directory:

   ```
   ORACLE_INSTANCE/config/OHS/component_name/moduleconf
   ```

   Add the following directives inside the `<VirtualHost>` tags:

   ```
   # 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>
   ```

   The `soa_vh.conf` file will appear as it does in Example 10–1.

2. Restart Oracle HTTP Server on WEBHOST1 and WEBHOST2:

   ```
   WEBHOST1> ORACLE_INSTANCE/bin/opmnctl restartproc ias-component=ohs1
   WEBHOST2> ORACLE_INSTANCE/bin/opmnctl restartproc ias-component=ohs2
   ```

10.3.5 Validating Access Through Oracle HTTP Server

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. Verify URLs to ensure that appropriate routing and failover is working from the HTTP Server to the BPM Suite Components.

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

To verify the URLs:

1. While WLS_SOA2 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`

10.4 Backing Up the Oracle BPM Configuration

After you have verified that the extended domain is working, back up the domain
configuration. This is a quick backup for the express purpose of immediate restore in
case of failures in future procedures. Back up the configuration to the local disk. This
backup can be discarded once you have completed the enterprise deployment. Once
you have completed the enterprise deployment, you can initiate the regular
deployment-specific backup and recovery process.

For information about backing up the environment, see "Backing Up Your
Environment" in the *Oracle Fusion Middleware Administrator's Guide*. For information
about recovering your information, see "Recovering Your Environment" in the *Oracle
Fusion Middleware Administrator's Guide*.

To back up the domain configuration:

1. Back up the Web tier:

   a. Shut down the instance using `opmnctl`.

      `ORACLE_INSTANCE/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_INSTANCE/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 are located in the following directory:

   `ASERVER_HOME`

   To back up the Administration Server run the following command on SOAHOST1:

   `tar -cvpf edgdomainback.tar ASERVER_HOME`

> **Note:** Back up *ORACLE_HOME* if any changes are made to the
> XEngine configuration that is part of your B2B setup. These files are
> located in the following directory:
>
> *ORACLE_HOME*/soa/thirdparty/edifecs/XEngine
>
> To back up *ORACLE_HOME*:
>
> `tar -cvpf fmwhomeback.tar MW_HOME`

11

Extending a SOA Domain to Oracle Service Bus

This chapter describes the procedures for extending the domain to include Oracle Service Bus.

This chapter contains the following sections:

- Section 11.1, "Overview of Adding Oracle Service Bus to a SOA Domain"
- Section 11.2, "Enabling VIP5 on SOAHOST1 and VIP6 on SOAHOST2"
- Section 11.3, "Running the Configuration Wizard on SOAHOST1 to Extend a SOA Domain to Include Oracle Service Bus"
- Section 11.4, "Disabling Host Name Verification for the WLS_OSBn Managed Servers"
- Section 11.5, "Configuring Oracle Coherence for the Oracle Service Bus Result Cache"
- Section 11.6, "Configuring a Default Persistence Store for Transaction Recovery"
- Section 11.7, "Propagating the Domain Configuration to the Managed Server Directory in SOAHOST1 and to SOAHOST2"
- Section 11.8, "Starting the Oracle Service Bus Servers"
- Section 11.9, "Validating the WLS_OSB Managed Servers"
- Section 11.10, "Configuring Oracle HTTP Server for the WLS_OSBn Managed Servers"
- Section 11.11, "Setting the Front End HTTP Host and Port for OSB_Cluster"
- Section 11.12, "Validating Access Through Oracle HTTP Server"
- Section 11.13, "Enabling High Availability for Oracle DB, File and FTP Adapters"
- Section 11.14, "Configuring Server Migration for the WLS_OSB Servers"
- Section 11.15, "Backing Up the Oracle Service Bus Configuration"

11.1 Overview of Adding Oracle Service Bus to a SOA Domain

This section provides an overview of adding Oracle Service bus to an SOA domain. Table 11–1 lists and describes to high-level steps for extending a SOA domain for Oracle Service Bus.

Table 11–1   Steps for Extending a SOA Domain to Include Oracle Service Bus

Step	Description	More Information
Enable VIP5 on SOAHOST1 and VIP6 on SOAHOST2	Enable a virtual IP mapping for each of these hostnames on the two SOA Machines.	Section 11.2, "Enabling VIP5 on SOAHOST1 and VIP6 on SOAHOST2"
Run the Configuration Wizard to Extend the Domain	Extend the SOA domain to contain Oracle Service Bus components	Section 11.3, "Running the Configuration Wizard on SOAHOST1 to Extend a SOA Domain to Include Oracle Service Bus"
Disable Host Name Verification for the WLS_OSBn Managed Server	If you have not set up the appropriate certificates for hostname verification between the Administration Server, Managed Servers, and Node Manager, disable host name verification.	Section 11.4, "Disabling Host Name Verification for the WLS_OSBn Managed Servers"
Configure Oracle Coherence for the Oracle Service Bus Result Cache	Use unicast communication for the Oracle Service Bus result cache.	Section 11.5, "Configuring Oracle Coherence for the Oracle Service Bus Result Cache"
Configure a Default Persistence Store for Transaction Recovery	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.	Section 11.6, "Configuring a Default Persistence Store for Transaction Recovery"
Propagate the Domain Configuration to the Managed Server Directory in SOAHOST1 and to SOAHOST2	Oracle Service Bus requires some updates to the WebLogic Server start scripts. Propagate these changes using the pack and unpack commands.	Section 11.7, "Propagating the Domain Configuration to the Managed Server Directory in SOAHOST1 and to SOAHOST2"
Start the Oracle Service Bus Servers	Oracle Service Bus servers extend an already existing domain. As a result, the Administration Server and respective Node Managers are already running in SOAHOST1 and SOAHOST2.	Section 11.8, "Starting the Oracle Service Bus Servers"
Validate the WLS_OSB Managed Servers	Verify that the server status is reported as Running in the Admin Console and access URLs to verify status of servers.	Section 11.9, "Validating the WLS_OSB Managed Servers"
Configuring Oracle HTTP Server for the WLS_OSBn Managed Servers	To enable Oracle HTTP Server to route to Oracle Service Bus console and Oracle Service Bus service, set the WebLogicCluster parameter to the list of nodes in the cluster.	Section 11.10, "Configuring Oracle HTTP Server for the WLS_OSBn Managed Servers"
Set the Front End HTTP Host and Port for OSB_Cluster	Set the front end HTTP host and port for Oracle WebLogic Server cluster.	Section 11.11, "Setting the Front End HTTP Host and Port for OSB_Cluster"

Table 11–1   (Cont.) Steps for Extending a SOA Domain to Include Oracle Service Bus

Step	Description	More Information
Validating Access Through Oracle HTTP Server	Verify that the server status is reported as Running.	Section 11.12, "Validating Access Through Oracle HTTP Server"
Enable High Availability for Oracle File and FTP Adapters	Make Oracle File and FTP Adapters highly available for outbound operations using the database mutex locking operation.	Section 11.13, "Enabling High Availability for Oracle DB, File and FTP Adapters"
Configure Server Migration for the WLS_OSB Servers	The high availability architecture for an Oracle Service Bus system uses server migration to protect some singleton services against failures.	Section 11.14, "Configuring Server Migration for the WLS_OSB Servers"

11.1.1 Prerequisites for Extending the SOA Domain to Include Oracle Service Bus

Before extending the current domain, ensure that your existing deployment meets the following prerequisites:

- Back up the installation - If you have not yet backed up the existing Fusion Middleware Home and domain, Oracle recommends backing it up now.

  To back up the existing Fusion Middleware Home and domain run the following command on SOAHOST1:

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

- You have installed WL_HOME and MW_HOME (binaries) on a shared storage and they are available from SOAHOST1 and SOAHOST2.

- You have already configured Node Manager, Admin Server, SOA Servers and WSM Servers as described in previous chapters to run a SOA system. You have already configured Server migration, transaction logs, coherence, and all other configuration steps for the SOA System.

11.2 Enabling VIP5 on SOAHOST1 and VIP6 on SOAHOST2

The SOA domain uses virtual hostnames as the listen addresses for the Oracle Service Bus managed servers. Enable a virtual IP mapping for each of these hostnames on the two SOA Machines, (VIP5 on SOAHOST1 and VIP6 on SOAHOST2), and correctly resolve the virtual hostnames in the network system used by the topology (either by DNS Server, hosts resolution).

To enable the virtual IP, follow the steps described in Section 3.5, "Enabling Virtual IP Addresses for Administration Servers." These virtual IPs and VHNs are required to enable server migration for the Oracle Service Bus Servers. Server migration must be configured for the Oracle Service Bus Cluster for high availability purposes. Refer to Chapter 9, "Server Migration" for more details on configuring server migration for the Oracle Service Bus servers.

11.3 Running the Configuration Wizard on SOAHOST1 to Extend a SOA Domain to Include Oracle Service Bus

In this step, you extend the domain created in Chapter 9, "Extending the Domain for SOA Components" to contain Oracle Service Bus components. The steps reflected in this section would be very similar if Oracle Service Bus was extending a domain containing only an Admin Server and a WSM-PM Cluster, but some of the options, libraries and components shown in the screens could vary.

To extend the domain for Oracle Service Bus:

1. Change directory to the location of the Configuration Wizard. This is within the Oracle Service Bus directory. (All database instances should be up.)

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

2. Start the Configuration Wizard.

    ```
    ./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
    ```

    Click **Next**.

5. In the Select Extension Source screen, select **Extend my domain automatically to support the following added products** and select the following products (the components required by Oracle SOA and Oracle WSM Policy Manager should already be selected and grayed out):

    - Oracle Service Bus OWSM Extension - 11.1.1.6 [osb]
    - Oracle Service Bus - 11.1.1.0 [osb]
    - WebLogic Advance Web Services JAX-RPC Extension

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

    - Select the select the **OSB JMS reporting Provider** schema.
    - For the Oracle RAC configuration for component schemas, select **Convert to GridLink**

    Click **Next**. The Configure Gridlink RAC Component Schema screen appears.

7. In the Configure Gridlink RAC Component Schema screen accept the values for the data sources that are already present in the domain and click **Next**.

8. In the Test JDBC Component Schema screen, verify that the Oracle Service Bus JMS reporting datasources are correctly verified and click **Next**.

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

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

    Click **Next**.

10. In the Select JMS Distributed Destination Type screen leave the pre-existing JMS System Resources as they are and Select **UDD** from the drop down list for **WseeJMSMOdule** and **JmsResources**.

Click **Next**.

11. In the Configure Managed Servers screen, add the required managed servers for Oracle Service Bus.

    a. Select the automatically created server and click **Rename** to change the name to WLS_OSB1.

    b. Click **Add** to add another new server and enter WLS_OSB2 as the server name.

    c. Give servers WLS_OSB1 and WLS_OSB2 the attributes listed in Table 11–2.

In the end, the list of managed servers should match Table 11–2.

Click **Next**.

Table 11–2   Managed Servers

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled
WLS_SOA1(*)	SOAHOST1VHN1	8001	n/a	No
WLS_SOA2(*)	SOAHOST2VHN1	8001	n/a	No
WLS_WSM1	SOAHOST1	7010	n/a	No
WLS_WSM2	SOAHOST2	7010	n/a	No
WLS_OSB1	SOAHOST1VHN2	8011	n/a	No
WLS_OSB2	SOAHOST2VHN2	8011	n/a	No

12. In the Configure Clusters screen, add the Oracle Service Bus cluster (leave the present cluster as they are):

Table 11–3   Clusters

Name	Cluster Messaging Mode	Multicast Address	Multicast Port	Cluster Address
SOA_Cluster(*)	unicast	n/a	n/a	SOAHOST1VHN1:8001, SOAHOST2VHN1:8001
WSM-PM_Cluster	unicast	n/a	n/a	Leave it empty.
OSB_Cluster	unicast	n/a	n/a	SOAHOST1VHN2:8011, SOAHOST2VHN2:8011

(*) - if you are extending a SOA domain

Click **Next**.

> **Note:** For asynch request/response interactions over direct binding, the SOA composites must provide their jndi provider URL for the invoked service to look up the beans for callback.
>
> If soa-infra configuration 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 configuration property `JndiProviderURL/SecureJndiProviderURL` can be used for the same purpose if explicitly set by users.

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

- SOA_Cluster - If you are extending a SOA domain.
  - WLS_SOA1
  - WLS_SOA2
- WSM-PM_Cluster:
  - WLS_WSM1
  - WLS_WSM2
- OSB_Cluster:
  - WLS_OSB1
  - WLS_OSB2

  Click **Next**.

14. Confirm that the following entries appear:

Table 11–4   Machines

Name	Node Manager Listen Address
SOAHOST1	SOAHOST1
SOAHOST2	SOAHOST2
ADMINHOST	localhost

Leave all other fields to their default values.

Click **Next**.

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

- ADMINHOST:
  - AdminServer
- SOAHOST1
  - WLS_SOA1 (if extending a SOA domain)
  - WLS_WSM1
  - WLS_OSB1
- SOAHOST2:

    – WLS_SOA2 (if extending a SOA domain)

    – WLS_WSM2

    – WLS_OSB2

Click **Next**.

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

- Target **usermessagingserver** and **usermessagingdriver-email** only to **SOA_Cluster**. (The usermessaging-xmpp, usermessaging-smpp, and usermessaging-voicexml applications are optional.)

- Target the **oracle.sdp.***, and **oracle.soa.*** libraries only to **SOA_Cluster**.

- Target the **oracle.rules.*** library only to **AdminServer** and **SOA_Cluster**.

- Target the **wsm-pm** application only to **WSM-PM_Cluster**.

- Target all Transport Provider Deployments to both the **OSB_Cluster** and the **AdminServer**.

Click **Next**.

For information on targeting applications and resources, see Appendix B, "Targeting Applications and Resources to Servers."

17. In the Target Services to Clusters or Servers screen:

- Target **mds-owsm** only to **WSM-PM_Cluster** and **AdminServer**.

- Target **mds-soa** only to **SOA_Cluster**.

Click **Next**.

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

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

Select **Direct-write** policy for all stores.

Click **Next**.

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

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

21. Restart the Administration Server for this configuration to take effect.

11.4 Disabling Host Name Verification for the WLS_OSB*n* Managed Servers

For the enterprise deployment described in this guide, you set up the appropriate certificates to authenticate the different nodes with the Administration Server after you have completed the procedures to extend the domain for Oracle SOA Suite. You must disable the host name verification for the WLS_OSB1 and WLS_OSB2 managed servers to avoid errors when managing the different WebLogic Server instances. For more information, see Section 8.4.8, "Disabling Host Name Verification."

You enable host name verification again once the enterprise deployment topology configuration is complete. For more information, see Section 13.3, "Enabling Host Name Verification Certificates for Node Manager in SOAHOST1."

11.5 Configuring Oracle Coherence for the Oracle Service Bus Result Cache

By default, result caching uses multicast communication. Oracle recommends using unicast communication for the Oracle Service Bus result cache. Additionally, Oracle recommends separating port ranges for Coherence clusters used by different products. The ports for the Oracle Service Bus result cache Coherence cluster should be different from the Coherence cluster used for SOA.

To enable unicast for the Oracle Service Bus result cache Coherence infrastructure:

1. Log into Oracle WebLogic Server Administration Console. In the Change Center, click Lock & Edit.

2. In the Domain Structure window, expand the Environment node.

3. Click **Servers**.

4. Click the name of the server (represented as a hyperlink) in the Name column of the table. The settings page for the selected server appears.

5. Click the Server Start tab.

6. Enter the following into the **Arguments** field for WLS_OSB1 on a single line, with no carriage returns:

   ```
   -DOSB.coherence.localhost=soahost1vhn2 -DOSB.coherence.localport=7890
   -DOSB.coherence.wka1=soahost1vhn2 -DOSB.coherence.wka1.port=7890
   -DOSB.coherence.wka2=soahost2vhn2 -DOSB.coherence.wka2.port=7890
   ```

   For WLS_OSB2, enter the following on a single line, no carriage returns:

   ```
   -DOSB.coherence.localhost=soahost2vhn2 -DOSB.coherence.localport=7890
   -DOSB.coherence.wka1=soahost1vhn2 -DOSB.coherence.wka1.port=7890
   -DOSB.coherence.wka2=soahost2vhn2 -DOSB.coherence.wka2.port=7890
   ```

   > **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.

7. Save and activate the changes. You must restart Oracle Service Bus servers for these changes take effect.

   > **Note:** The Coherence cluster used for Oracle Service Bus' result cache is configured above using port 7890. This port can be changed by specifying a different port (for example, 8089) with the following startup parameters:
   >
   > ```
   > -Dtangosol.coherence.wkan.port
   > -Dtangosol.coherence.localport
   > ```
   >
   > For more information about Coherence Clusters see the *Oracle Coherence Developer's Guide*.

8. Ensure that these variables are passed to the managed server correctly by checking the server's output log.

   Failure of the Oracle Coherence framework can prevent the result caching from working.

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

To set the location for the default persistence stores:

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. Perform Steps 2-5 for both WLS_OSB1, and WLS_OSB2 servers.

7. Click **Save** and **Active Changes**.

8.

> **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_OSB1 and WLS_OSB2 must be able to access this directory. This directory must also exist before you restart the servers.

11.7 Propagating the Domain Configuration to the Managed Server Directory in SOAHOST1 and to SOAHOST2

Oracle Service Bus requires some updates to the WebLogic Server start scripts. Propagate these changes using the pack and unpack commands.

> **Note:**   If you configure server migration before unpacking (extending the domain to include OSB), the `wlsifconfig` script is overwritten and must be restored from the copy that the `-overwrite` option provides.

Prerequisite

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

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

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

    ```
    cd ORACLE_COMMON_HOME/common/bin

    ./pack.sh -managed=true -domain=ORACLE_BASE/admin/
    domain_name/aserver/domain_name -template=soadomaintemplateExtOSB.jar
     -template_name=soa_domain_templateExtOSB
    ```

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

    ```
    ./unpack.sh -domain=ORACLE_BASE/admin/domain_name/mserver/domain_name
    -overwrite_domain=true -template=soadomaintemplateExtOSB.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.

3. Copy the template to SOAHOST2 run the following commands on SOAHOST1:

    ```
    cd ORACLE_COMMON_HOME/common/bin

    scp soadomaintemplateExtOSB.jar oracle@SOAHOST2:/ ORACLE_COMMON_HOME/common/bin
    ```

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

    ```
    cd ORACLE_COMMON_HOME/common/bin

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

> **Note:** The configuration steps provided in this enterprise deployment topology are documented with the assumption that a local (per node) domain directory is used for each managed server.

11.8 Starting the Oracle Service Bus Servers

Since Oracle Service Bus servers extend an already existing domain it is assumed that the Administration Server and respective Node Managers are already running in SOAHOST1 and SOAHOST2.

To start the added the WLS_OSB servers:

1. Log into the Oracle WebLogic Server Administration Console at:

   ```
   http://ADMINVHN:7001/console
   ```

2. In the Domain Structure window, expand the **Environment** node, then select **Servers**.

   The Summary of Servers page appears.

3. Click the **Control** tab.

4. Select **WLS_OSB1** from the **Servers** column of the table.

5. Click **Start**. Wait for the server to come up and check that its status is reported as **RUNNING** in the Administration Console.

6. Repeat steps 2 through 5 for WLS_OSB2.

11.9 Validating the WLS_OSB Managed Servers

Validate the WLS_OSB managed servers using the Oracle WebLogic Server Administration Console and by accessing URLs.

To validate the WLS_OSB managed server:

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 16.14, "Troubleshooting the Topology in an Enterprise Deployment" for possible causes.

2. Access the following URL to verify status of WLS_OSB1:

   ```
   http://SOAHOST1VHN2:8011/sbinspection.wsil
   ```

   With the default installation, this should be the HTTP response:

Figure 11–1   HTTP Response

```
<?xml version="1.0" encoding="UTF-8" ?>
- <ins:inspection xmlns:ins="http://schemas.xmlsoap.org/ws/2001/10/inspection/">
  - <ins:link referencedNamespace="http://schemas.xmlsoap.org/ws/2001/10/inspection/" location="http://:
    <ins:abstract>default</ins:abstract>
    <ins:abstract>LinkType: Project</ins:abstract>
  </ins:link>
</ins:inspection>
```

3. Access the following URL:

   ```
   http://SOAHOST1VHN2:8011/alsb/ws/_async/AsyncResponseServiceJms?WSDL
   ```

With the default installation, this should be the HTTP response:

Figure 11–2 HTTP Response

4. Access the equivalent URLs for:

   ```
   http://SOAHOST2VHN2:8011/
   ```

5. Verify also the correct deployment of the Oracle Service Bus console to the Administration Server by accessing the following URL:

   ```
   http://ADMINHOSTVHN:7001/sbconsole/
   ```

   The Oracle Service Bus console should appear with no errors.

11.10 Configuring Oracle HTTP Server for the WLS_OSBn Managed Servers

To enable Oracle HTTP Server to route to Oracle Service Bus console and Oracle Service Bus service you must set the WebLogicCluster parameter to the list of nodes in the cluster.

If you use a virtual server for administration purposes, the /sbconsole defines routing in the context of the virtual server in the admin_vh.conf file. Similarly, add the rest of the Oracle Service Bus URLs to the osb_vh.conf file.

Start the context paths for the HTTP proxy services with a common name, such as /osb/project-name/folder-name/proxy-service-name to facilitate the routing in Oracle HTTP Server for all the proxy services.

To set the parameter:

1. On WEBHOST1 and WEBHOST2, add directives to the osb_vh.conf file located in the following directory:

   ```
   ORACLE_BASE/admin/instance_name/config/OHS/component_name/moduleconf
   ```

   Note that this assumes you created the osb_vh.conf file using the instructions in Section 7.6, "Defining Virtual Hosts."

   Add the following directives inside the <VirtualHost> tags:

   ```
   <Location /sbinspection.wsil >
       SetHandler weblogic-handler
       WebLogicCluster SOAHOST1VHN2:8011,SOAHOST2VHN2:8011
       WLProxySSL ON
       WLProxySSLPassThrough ON
   </Location>

   <Location /sbresource >
   ```

```
        SetHandler weblogic-handler
        WebLogicCluster SOAHOST1VHN2:8011,SOAHOST2VHN2:8011
        WLProxySSL ON
        WLProxySSLPassThrough ON
</Location>

<Location /osb >
        SetHandler weblogic-handler
        WebLogicCluster SOAHOST1VHN2:8011,SOAHOST2VHN2:8011
        WLProxySSL ON
        WLProxySSLPassThrough ON
</Location>

<Location /alsb >
        SetHandler weblogic-handler
        WebLogicCluster SOAHOST1VHN2:8011,SOAHOST2VHN2:8011
        WLProxySSL ON
        WLProxySSLPassThrough ON
</Location>
```

The `osb_vh.conf` file will appear as it does in Example 11–1.

2. Add the following entry to the `admin_vh.conf` file within the `<VirtualHost>` tags:

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

The `admin_vh.conf` file will appear as it does in Example 11–2.

3. 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
```

Example 11–1   osb_vh.conf file

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

<Location /sbinspection.wsil >
    SetHandler weblogic-handler
    WebLogicCluster SOAHOST1VHN2:8011,SOAHOST2VHN2:8011
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>

<Location /sbresource >
    SetHandler weblogic-handler
    WebLogicCluster SOAHOST1VHN2:8011,SOAHOST2VHN2:8011
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>
```

```
<Location /osb >
    SetHandler weblogic-handler
    WebLogicCluster SOAHOST1VHN2:8011,SOAHOST2VHN2:8011
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>

<Location /alsb >
    SetHandler weblogic-handler
    WebLogicCluster SOAHOST1VHN2:8011,SOAHOST2VHN2:8011
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>
</VirtualHost>
```

Example 11–2   admin_vh.conf file

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

<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>

<Location /sbconsole >
    SetHandler weblogic-handler
    WebLogicHost ADMINVHN
    WeblogicPort 7001
   </Location>
</VirtualHost>
```

11.11 Setting the Front End HTTP Host and Port for OSB_Cluster

Set the front end HTTP host and port for Oracle WebLogic Server cluster using the WebLogic Server Administration Console.

To set the front end host and port:

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

2. In the left pane, select **Environment** and then **Clusters**.

3. Select the **OSB_Cluster**.

4. Select **HTTP**.

5. Set the values for the following:

   - Frontend Host: **osb.mycompany.com**

   - Frontend HTTP Port: **80**

   - Frontend HTTPS Port: **443**

   > **Note:** Make sure this address is correct and available (the load balancing router is up). An incorrect value, for example, http:// in the address, or trailing / in the hostname, may prevent the SOA system from being accessible even when using the virtual IPs to access it.

   Click **Save**.

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

11.12 Validating Access Through Oracle HTTP Server

Since you have already set the cluster address for the OSB_Cluster, the Oracle Service Bus URLs can only be verified once Oracle HTTP Server has been configured to route the Oracle Service Bus context URLs to the WebLogic Servers. Verify the URLs to ensure that appropriate routing and failover is working from the HTTP Server to the Oracle Service Bus components.

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

To verify the URLs:

1. While WLS_OSB1 is running, stop WLS_OSB2 using the Oracle WebLogic Server Administration Console.

2. Access WebHost1:7777/sbinspection.wsil and verify the HTTP response as indicated in Section 11.9, "Validating the WLS_OSB Managed Servers."

3. Start WLS_OSB2 from the Oracle WebLogic Server Administration Console.

4. Stop WLS_OSB1 from the Oracle WebLogic Server Administration Console.

5. Access WebHost1:7777/sbinspection.wsil and verify the HTTP response as indicated in section Section 11.9, "Validating the WLS_OSB Managed Servers."

   > **Note:** Since a front end URL has been set for the OSB_Cluster, the requests to the urls result in a re-route to the LBR, but in all cases it should suffice to verify the appropriate mount points and correct failover in Oracle HTTP Server.

6. Verify this URLs using your load balancer address:

```
http://osb.mycompany.com:80/sbinspection.wsil
```

11.13 Enabling High Availability for Oracle DB, File and FTP Adapters

Oracle SOA Suite and Oracle Service Bus use the same database and File and FTP JCA adapters. You create the required database schemas for these adapters when you use the Oracle Repository Creation Utility for SOA. The required configuration for the adapters is described in section Section 9.8.1, "Enabling High Availability for Oracle File and FTP Adapters." The DB adapter does not require any configuration at the WebLogic Server resource level. If you are configuring Oracle Service Bus as an extension of a SOA domain, you do not need to add to the configuration already performed for the adapters.

If you are deploying Oracle Service Bus as an extension to a WSM-PM and Admin Server domain, do the following:

- Run RCU to seed the Oracle Service Bus database with the required adapter schemas (Select **SOA Infrastructure**, and **SOA and BAM Infrastructure** in RCU).

- Perform the steps in, and the steps reflected in Section 9.8.1, "Enabling High Availability for Oracle File and FTP Adapters."

11.14 Configuring Server Migration for the WLS_OSB Servers

The high availability architecture for an Oracle Service Bus system uses server migration to protect some singleton services against failures. For more information on whole server migration, see Oracle Fusion Middleware Using Clusters for Oracle WebLogic Server.

The WLS_OSB1 managed server is configured to be restarted on SOAHOST2 in case of failure, and the WLS_OSB2 managed server is configured to be restarted on SOAHOST1 in case of failure. For this configuration the WLS_OSB1 and WLS_OSB2 servers listen on specific floating IPs that are failed over by WLS Server Migration.

Table Table 11–5 lists the high-level steps for configuring server migration for the WLS_OSB server.

Table 11–5   Steps for Configuring Server Migration for the WLS_OSB Servers

Step	Description	More Information
Set Up the User and Tablespace for the Server Migration Leasing Table	Set Up the User and Tablespace for the Server Migration Leasing Table. If a tablespace has already been set up for SOA, this step is not required.	Section 11.14.1, "Setting Up the User and Tablespace for the Server Migration Leasing Table"
Edit the Node Manager's Properties File	Edit the Node Manager properties file on the two nodes where the servers are running.	Section 11.14.2, "Editing the Node Manager's Properties File"

Table 11–5 (Cont.) Steps for Configuring Server Migration for the WLS_OSB Servers

Step	Description	More Information
Set Environment and Superuser Privileges for the wlsifconfig.sh Script	Set the environment and superuser privileges for the wlsifconfig.sh script.	Section 11.14.3, "Setting Environment and Superuser Privileges for the wlsifconfig.sh Script"
Configure Server Migration Targets	Configure cluster migration targets, set the DataSourceForAutomaticMigration property to true.	Section 11.14.4, "Configuring Server Migration Targets"
Validate Server Migration	Verify that Server Migration is working properly.	Section 11.14.5, "Validating Server Migration"

11.14.1 Setting Up the User and Tablespace for the Server Migration Leasing Table

Set up the User and tablespace for the server migration table using SQL*Plus.

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 password;
SQL> grant create table to leasing;
SQL> grant create session to leasing;
SQL> alter user leasing default tablespace leasing;
SQL> alter user leasing quota unlimited on LEASING;
```

3. Create the leasing table using the `leasing.ddl` script as follows:

    a. Copy the *leasing.ddl* file located in the following directory to your database node:

    ```
    WL_HOME/server/db/oracle/920
    ```

    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;
    ```

11.14.2 Editing the Node Manager's Properties File

Edit the Node Manager properties file on the two nodes where the servers are running. The `nodemanager.properties` file is located in the following directory:

```
WL_HOME/common/nodemanager
```

Add the following properties to enable server migration to work properly:

■   `Interface`

`Interface=eth0`

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

  NetMask=255.255.255.0

  This property specifies the net mask for the interface for the floating IP.

- UseMACBroadcast

  UseMACBroadcast=true

  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 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 (Linux) or Interface="Local Area Connection" (Windows)
NetMask=255.255.255.0
UseMACBroadcast=true
...
```

11.14.3 Setting Environment and Superuser Privileges for the wlsifconfig.sh Script

Set the environment and superuser privileges for the `wlsifconfig.sh` script.

To set the environment and superuser privileges:

1. Ensure that the PATH environment variable includes the files listed in Table 11–6.

Table 11–6   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.domain	*WL_HOME*/common/nodemanager

2. Grant sudo privilege to the WebLogic user ('oracle') with no password restriction, and grant execute privilege on the */sbin/ifconfig* and */sbin/arping* binaries.

   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.

   On Windows, the script is named `wlsifconfig.cmd` and it can be run with the administrator privilege.

> **Note:** Ask the system administrator for the sudo and system rights as appropriate to this step.

3. 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`.

To grant sudo privilege to the WebLogic user ('oracle') with no password restriction, and grant execute privilege on the */sbin/ifconfig* and */sbin/arping* binaries:

```
Defaults:oracle !requiretty
oracle ALL=NOPASSWD: /sbin/ifconfig,/sbin/arping
```

11.14.4 Configuring Server Migration Targets

Configure cluster migration by setting the `DataSourceForAutomaticMigration` property to `true`.

To configure migration targets 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 (**OSB_Cluster**) in the Name column of the table.

4. Click the **Migration** tab.

5. In the Change Center Click **Lock & Edit**.

6. In the **Available** field, select the machine to which to allow migration and click the right arrow. In this case, select **SOAHOST1** and **SOAHOST2**.

7. Select the data source to be used for automatic migration. In this case select the leasing data source and click **Save**.

8. Click **Activate Changes**.

9. Set the candidate machines for server migration for WLS_OSB1 and WLS_OSB2:

Set the candidate machines for server migration for WLS_OSB1 only. WLS_OSB2 does not use server migration:" need to be reformulate. We need configure for both OSB1 and OSB2, because OSB servers are identical. We used this note for BAM, because BAM1 is not identical with BAM2

a. In the 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 **SOAHOST2** for **WLS_OSB1**. Select **SOAHOST1** for **WLS_OSB2**.

e. Select **Automatic Server Migration Enabled** and click **Save**.

This enables the Node Manager to start a failed server on the target node automatically.

    f. Click **Activate Changes**.

    g. Restart the Administration Server and the WLS_OSB1 server.

To restart the Administration Server, use the procedure in Section 8.4.3, "Starting the Administration Server on SOAHOST1."

For information on targeting applications and resources, see Appendix B, "Targeting Applications and Resources to Servers."

11.14.5 Validating Server Migration

To validate that Server Migration is working properly:

To test from Node 1:

1. Force stop the WLS_OSB1 managed server.

```
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:

```
ps -ef | grep WLS_OSB1
```

Note: For Windows, the Managed Server can be terminated by using the taskkill command. For example:

```
taskkill /f /pid pid
```

Where *pid* is the process Id of the managed server.

To determine the process Id of the WLS_OSB1 Managed Server:

```
MW_HOME\jrockit_160_20_D1.0.1-2124\bin\jps -l -v
```

---

2. Watch the Node Manager console: you should see a message indicating that WLS_OSB1's floating IP has been disabled.

3. Wait for the Node Manager to try a second restart of WLS_OSB1. 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.

To test from Node 2:

1. Watch the local Node Manager console. After 30 seconds since the last try to restart WLS_OSB1on Node 1, Node Manager on Node 2 should prompt that the floating IP for WLS_OSB1 is being brought up and that the server is being restarted in this node.

2. Access the `sbinspection.wsil urlconsole` in the same IP.

To validate migration from the WebLogic Server 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.

Figure 11–3   Migration Status Screen in the Administration Console
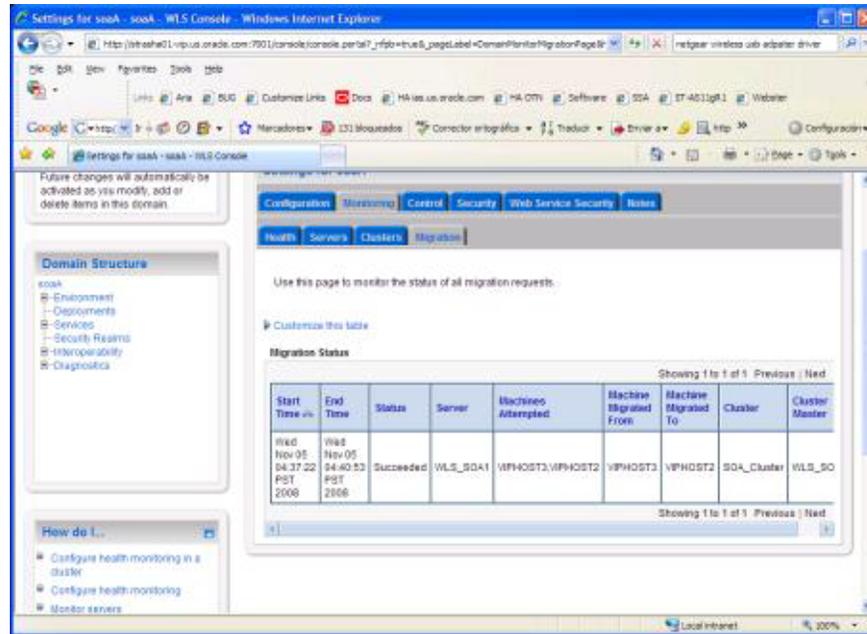
> **Note:**   After a server is migrated, to fail it back to its original node,
> stop the managed server from the Oracle WebLogic Administration
> Console and then start it again. The appropriate Node Manager starts
> the managed server on the machine to which it was originally
> assigned.

11.15 Backing Up the Oracle Service Bus Configuration

After you have verified that the extended domain is working, back up the domain
configuration. This is a quick backup for the express purpose of immediate restore in
case of failures in future procedures. Back up the configuration to the local disk. This
backup can be discarded once you have completed the enterprise deployment. Once
you have completed the enterprise deployment, you can initiate the regular
deployment-specific backup and recovery process.

For information about backing up the environment, see "Backing Up Your
Environment" in the *Oracle Fusion Middleware Administrator's Guide*. For information
about recovering your information, see "Recovering Your Environment" in the *Oracle
Fusion Middleware Administrator's Guide*.

To back up the domain configuration:

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 are located in the following directory:

```
ORACLE_BASE/admin/domain_name
```

To back up the Administration Server:

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

12

Extending the Domain to Include BAM

This chapter describes the procedures for extending the domain to include Oracle Business Activity Monitoring (BAM).

This chapter contains the following sections:

12.1 Overview of Adding BAM to a Domain

The BAM system is installed using the WL_HOME and ORACLE_HOME created in Chapter 8, "Creating a Domain for an Enterprise Deployment" 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.

12.2 Prerequisites for Extending the Domain to Include BAM

Before performing the steps in this section review the following prerequisites:

Back up the existing installation

If you have not yet backed up the existing Fusion Middleware Home and domain, back it up now.

To back up the existing Fusion Middleware Home and domain:

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

Attach the existing Oracle Homes to the Inventory in the BAMHOSTS

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.

To attach an *ORACLE_HOME* in shared storage to the local Oracle Inventory, use the following command on the BAMHOST:

```
cd ORACLE_COMMON_HOME/oui/bin/attachHome.sh
./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.

12.3 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. Enable a virtual IP (VIP4) mapping BAMHOST1VHN1 on BAMHOST1 and correctly resolve the BAMHOST1VHN1 hostname in the network system used by the topology (either by DNS Server, hosts resolution).

To enable the virtual IP, see Section 3.5, "Enabling Virtual IP Addresses for Administration Servers."

About Accessing an Oracle BAM Server Using the BAM Adapter

When accessing an Oracle BAM Server using the BAM Adapter (rmi), use the Virtual Hostname of the BAM Server (BAMHOST1VNH1) for the connection. SOAP requests come through HTTP, therefore, use the load balancer addresses when using the adapter.

12.4 Running the Configuration Wizard to Extend the Domain

In this step, you extend the domain created in Chapter 8, "Creating a Domain for an Enterprise Deployment" 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, you may choose to use a different database service specifically for BAM.

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

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

2. Start the Configuration Wizard:

   ```
   ./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 following WebLogic domain directory:

   ```
   ORACLE_BASE/admin/domain_name/aserver/domain_name
   ```

   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 Components Schema screen, do the following:

   - Select **BAM Schema**.

   - For the Oracle RAC configuration for component schemas, select **Convert to GridLink**.

   Click **Next**.

7. The Configure Gridlink RAC Component Schema screen appears (Figure 12–1).

Figure 12–1   Configure GridLink RAC Component Schema Screen

In this screen enter values for the following fields, specifying the connect information for the Oracle RAC database that was seeded with RCU:

– **Driver:** Select **Oracle's driver (Thin) for GridLinkConnections,Versions:10 and later**.

– **Service Name:** Enter the service name of the database using lowercase characters. For example:

```
soaedg.mycompany.com
```

– **Username:** Enter the database schema owner name of the corresponding component.

– **Password:** Enter the password for the database schema owner.

– Select **Enable FAN**

– Make sure **Enable SSL** is unchecked (alternatively if ssl is selected for ONS notifications to be encrypted, provide the appropriate wallet and wallet password).

– **Service listener**: Enter the SCAN address and port for the RAC database being used. You can identify this address by querying the appropriate parameter in the database using the TCP protocol:

```
SQL>show parameter remote_listener;

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

> **Note:** For Oracle Database 11*g* Release 1 (11.1), use the virtual IP and port of each database instance listener, for example:
>
> ```
> custdbhost1-vip.mycompany.com (port 1521)
> ```
>
> and
>
> ```
> custdbhost2-vip.mycompany.com (1521)
> ```
>
> For Oracle Database 10*g*, use multi data sources to connect to an Oracle RAC database. For information about configuring multi data sources see Appendix A, "Using Multi Data Sources with Oracle RAC."

- **ONS Host**: Enter the SCAN address for the Oracle RAC database and the ONS remote port as reported by the database:

  ```
  [orcl@db-scan1 ~]$ srvctl config nodeapps -s

  ONS exists: Local port 6100, remote port 6200, EM port 2016
  ```

  > **Note:** For Oracle Database 11*g* Release 1 (11.1), use the hostname and port of each database's ONS service, for example
  >
  > ```
  > custdbhost1.mycompany.com (port 6200)
  > ```
  >
  > and
  >
  > ```
  > custdbhost2.mycompany.com (6200)
  > ```

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 Select Optional Configuration screen, select the following:

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

   - **Deployments and Services**

   - **JMS File Store**

   - **JMS Distributed Destination**

     Click **Next**.

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 12–1. Do not modify the other servers that appear in this screen; leave them as they are.

Table 12–1   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 12–2. Do not modify the other clusters that display in this screen; leave them as they are.

Table 12–2   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 12–3   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**.

   ■ Target the **oracle.sdp.*** library only to **SOA_Cluster** and **BAM_Cluster**. Target the **oracle.soa.*** library only to **SOA_Cluster**.

   ■ Target the **oracle.rules.*** library to **SOA_Cluster**, **BAM_Cluster** and **Admin Server**.

   ■ **oracle.bam*** is targeted only to **BAM_Cluster**.

   Click **Next**.

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

   ■ Target **mds-owsm** to both **WSM-PM_Cluster** and **AdminServer**.

   ■ Target **mds-soa** to both **SOA_Cluster** and **AdminServer**.

   ■ Target **OraSDPMDatasource** to both **SOA_Cluster** and **BAM_Cluster**.

   Click **Next**.

   For information on targeting applications and resources, see Appendix B, "Targeting Applications and Resources to Servers."

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

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

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

20. Restart the Administration Server using the procedure in Section 8.4.3, "Starting the Administration Server on SOAHOST1."

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

To set the location for the default persistence store for WLS_BAM1:

1. Log into the Oracle WebLogic Server Administration Console.

2. Click **Lock & Edit**.

3. In the Domain Structure window, expand the **Environment** node and then click the **Servers** node. The Summary of Servers page appears.

4. 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.

5. Click the **Services** sub-tab.

6. In the **Default Store** section of the page, enter the path to the folder where the default persistent stores will store its data files. The directory structure of the path is as follows:

    ORACLE_BASE/admin/*domain_name*/*bam_cluster_name*/tlogs

7. Click **Save**.

8. Click **Activate Changes**.

> **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.

12.6 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.

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 & Edit**.

8. Change the targets for the modules as described in Table 12–4. You must change all of these components, as incorrect targeting can prevent the BAM system from starting.

Table 12–4   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**

   For information on targeting applications and resources, see Appendix B, "Targeting Applications and Resources to Servers."

12.7 Propagating the Domain Changes to the Managed Server Domain Directory

Propagate the domain configuration to BAMHOST1 and BAMHOST2 using the pack/unpack utility.

Note: If you configure server migration before unpacking (extending the domain to include BAM), the `wlsifconfig` script is overwritten and must be restored from the copy tha the `-overwrite` option provides.

To propagate the new domain configuration:

1. Make sure that a similar directory and shared storage configuration as SOAHOST2 is present in BAMHOST1 (described in Chapter 3, "Preparing the Network for an Enterprise Deployment"). BAMHOST1 and BAMHOST2 should have mounted the `MW_HOME` directory as created in Chapter 8, "Creating a Domain for an Enterprise Deployment."

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

   a. Run the following command:

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

   Note: Notice that this directory is available as mount point to the MW_HOME created in Chapter 8, "Creating a Domain for an Enterprise Deployment."

   ---

   b. Run the pack command:

   ```
   ./pack.sh -managed=true -domain=ORACLE_BASE/admin/
   ```

```
domain_name/aserver/domain_name -template=soadomaintemplateExtBAM.jar
 -template_name=soa_domain_templateExtBAM
```

3. Run the following command on SOAHOST1 to copy the template file created in the previous step to BAMHOST1.

```
scp soadomaintemplateExtBAM.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=soadomaintemplateExtBAM.jar
-app_dir=ORACLE_BASE/admin/domain_name/mserver/applications
```

5. Run the copy and unpack commands for BAMHOST2.

Note: The configuration steps provided in this enterprise deployment topology are documented with the assumption that a local (per node) domain directory is used for each managed server.

12.8 Disabling Host Name Verification for the WLS_BAM*n* Managed Servers

For the enterprise deployment described in this guide, you set up the appropriate certificates to authenticate the different nodes with the Administration Server after you have completed the procedures to extend the domain for Oracle SOA Suite. You must disable the host name verification for the WLS_BAM1 and WLS_BAM2 managed servers to avoid errors when managing the different WebLogic Server instances. For more information, see Section 8.4.8, "Disabling Host Name Verification."

You enable host name verification again once the enterprise deployment topology configuration is complete. For more information, see Section 13.3, "Enabling Host Name Verification Certificates for Node Manager in SOAHOST1."

12.9 Starting Node Manager on BAMHOST1 and BAMHOST2

Start Node manager using the setNMProps.sh script.

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: Use the StartScriptEnabled property to avoid class loading failures and other problems. For more information about StartScriptEnabled, see Section 16.14.5, "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 3, "Preparing the Network for an Enterprise Deployment," it is not required to run `setNMProps.sh` again. In this case, Node Manager has already been configured to use a 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
```

12.10 Starting the BAM System

Start the WLS_BAM1 managed server using the Oracle WebLogic Server Administration Console.

These instructions are based on the assumption 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.

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 5.2.3, "About 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.

12.11 Validating GridLink Data Sources

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

To validate the GridLink data sources configuration:

1. Log on to the Oracle WebLogic Administration Console.

2. In the **Domain Structure** tree, expand **Services**, and select **Data Sources**.

3. Click one of the new data sources.

4. Click the **Monitoring** tab and select one of the servers.

5. Click the **Statistics** tab and select one of the servers.

6. Click the **ONS** tab, and then click the **Testing** tab.

7. Select the server and click **Test ONS**.

   If both tests are successful, the configuration is correct. If the ONS test fails, verify that the ONS service is running in the RAC database nodes:

   ```
   orcl@db-scan1 ~]$ srvctl status scan_listener
   SCAN Listener LISTENER_SCAN1 is enabled
   SCAN listener LISTENER_SCAN1 is running on node db-scan1
   SCAN Listener LISTENER_SCAN2 is enabled
   SCAN listener LISTENER_SCAN2 is running on node db-scan2
   SCAN Listener LISTENER_SCAN3 is enabled
   SCAN listener LISTENER_SCAN3 is running on node db-scan2

   [orcl@db-scan1 ~]$ srvctl config nodeapps -s
   ONS exists: Local port 6100, remote port 6200, EM port 2016

   [orcl@db-scan1 ~]$ srvctl status nodeapps | grep ONS
   ONS is enabled
   ONS daemon is running on node: db-scan1
   ONS daemon is running on node: db-scan2
   ```

   Run the ONS test from every WebLogic server that uses the data source.

12.12 Configuring the BAM Web Applications to Use the BAM Server in BAMHOST1

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 3–2 in Section 3.4, "About IPs and Virtual IPs."

   ■ Click **Apply**.

   ■ Modify the listening port of the server using the Mbean browser:

      – 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.

      – Highlight **OracleBamServer** in the left navigation tree.

      – In the **BAMServer** drop-down menu on the top-right, select **System 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 steps 3 through 5.

12.13 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:

To enable Oracle HTTP Server to route to the SOA_Cluster:

1. On WEBHOST1 and WEBHOST2, add directives to the `soa_vh.conf` file located in the following directory:

   ORACLE_BASE/admin/*instance_name*/config/OHS/*component_name*/moduleconf

   Note that this assumes you created the `soa_vh.conf` file using the instructions in Section 7.6, "Defining Virtual Hosts."

   Add the following directives to the `soa_vh.conf` file within the `<VirtualHost>` tags.

```
# 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>
```

The `soa_vh.conf` file will appear as it does in Example 12–1.

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
```

Example 12–1   soa_vh.conf file

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

<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 /workflow/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>

# 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>
</VirtualHost>
```

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. 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.

12.14 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.

To verify the 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. You can not retrieve reports or data at this point 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.

12.15 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 Gridlink Data Source for leasing Using the Administration Console

- Step 3: Editing the Node Manager's Properties File

- Step 4: Setting Environment and Superuser Privileges for the wlsifconfig.sh Script

- Step 5: Enabling Host Name Verification for Node Manager in the BAMHOSTn Nodes and the Administration Server

- Step 6: Configuring Server Migration Targets

- Step 7: Testing 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 are not required, but you must target the corresponding server-migration/leasing datasources to the BAM Cluster.

12.15.1 Setting Up the User and Tablespace for the Server Migration Leasing Table

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 password;
SQL> grant create table to leasing;
SQL> grant create session to leasing;
SQL> alter user leasing default tablespace leasing;
SQL> alter user leasing quota unlimited on LEASING;
```

3. Create the leasing table using the `leasing.ddl` script 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;
   ```

12.15.2 Creating a Gridlink Data Source for leasing Using the Administration Console

To use the Oracle WebLogic Server Administration Console to create GridLink data source for the leasing table follow the instructions in section Section 14.3, "Creating a GridLink Data Source for Leasing Using the Administration Console."

12.15.3 Editing the Node Manager's Properties File

Edit the Node Manager properties file on the two nodes where the servers are running. The `nodemanager.properties` file is located in the following directory:

WL_HOME/common/nodemanager

Add the following properties to enable server migration to work properly:

- `Interface`

  `Interface=eth0`

  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

  `NetMask=255.255.255.0`

  This property specifies the net mask for the interface for the floating IP.

- UseMACBroadcast

  `UseMACBroadcast=true`

  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 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 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.

12.15.4 Setting Environment and Superuser Privileges for the wlsifconfig.sh Script

Set the environment and superuser privileges for the `wlsifconfig.sh` script.

Ensure that the PATH environment variable includes the files listed in Table 12–5.

Table 12–5   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.domain	*WL_HOME*/common/nodemanager

Grant sudo privilege to the WebLogic user ('oracle') with no password restriction, and grant execute privilege on the */sbin/ifconfig* and */sbin/arping* binaries.

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.

> **Note:**   Ask the system administrator for the sudo and system rights as appropriate to this step.

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`.

To grant sudo privilege to the WebLogic user ('oracle') with no password restriction, and grant execute privilege on the */sbin/ifconfig* and */sbin/arping* binaries:

```
oracle ALL=NOPASSWD: /sbin/ifconfig,/sbin/arping
```

12.15.5 Enabling Host Name Verification for Node Manager in the BAMHOST*n* Nodes and the Administration Server

Enable host name verification certificates between Node Manager in the BAMHOST*n* nodes and the Administration Server. To enable host name verification certificates, see Section 13.3, "Enabling Host Name Verification Certificates for Node Manager in SOAHOST1."

12.15.6 Configuring Server Migration Targets

To configure cluster migration targets, set the `DataSourceForAutomaticMigration` property to true.

To configure migration targets 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. Click **Lock & Edit**.

6. In the **Available** field, select the machine to which to allow migration and click the right arrow. In this case, select **BAMHOST1** and **BAMHOST2**.

7. Select the **data source** to be used for automatic migration. In this case select the **leasing** data source.

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

9. Set the candidate machines for server migration for WLS_BAM1 only. 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** and click **Save**.

   This enables the Node Manager to start a failed server on the target node automatically.

   f. Click **Activate Changes**.

   g. Restart the Administration Server and the WLS_BAM1 server.

   To restart the Administration Server, use the procedure in Section 8.4.3, "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 is different from the configuration if the server is migrated automatically.

12.15.7 Testing Server Migration

To verify that Server Migration is working properly:

To test from Node 1:

1. Kill the WLS_BAM1 managed server.

   ```
   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.

To test 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.

To validate migration from 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.

12.16 Applying Configuration Changes BAM components in a BAM Cluster

If you are using Oracle BAM in a clustered environment, you must make the changes you made in Oracle Enterprise Manager on one node to all nodes. In addition, consider the following when making configuration changes to BAM Server in a BAM Enterprise Deployment Topology:

Since you are using server migration, the BAM Server is moved to a different node's domain directory. You must 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
   ```

12.17 Backing Up the BAM Configuration

After you have verified that the extended domain is working, back up the domain configuration. This is a quick backup for the express purpose of immediate restore in case of failures in future procedures. Back up the configuration to the local disk. This backup can be discarded once you have completed the enterprise deployment. Once you have completed the enterprise deployment, you can initiate the regular deployment-specific backup and recovery process.

For information about backing up the environment, see "Backing Up Your Environment" in the *Oracle Fusion Middleware Administrator's Guide*. For information about recovering your information, see "Recovering Your Environment" in the *Oracle Fusion Middleware Administrator's Guide*.

To back up the domain configuration:

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 are located in the following directory:

   ```
   ORACLE_BASE/ admin/domain_name
   ```

   To back up the Administration Server:

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

13

Setting Up Node Manager for an Enterprise Deployment

This chapter describes how to configure Node Manager according to the Enterprise Deployment recommendations.

This chapter includes the following sections:

- Section 13.1, "Overview of the Node Manager"
- Section 13.2, "Changing the Location of Node Manager Log"
- Section 13.3, "Enabling Host Name Verification Certificates for Node Manager in SOAHOST1"
- Section 13.4, "Starting the Node Manager on SOAHOST1"
- Section 13.5, "Enabling Host Name Verification Certificates for the Node Manager in SOAHOST2"
- Section 13.6, "Starting Node Manager on SOAHOST2"
- Section 13.7, "Configuring WebLogic Servers to Use the Custom Keystores"

13.1 Overview of the Node Manager

The Node Manager enables you to start and stop the Administration Server and the managed servers.

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 host name changes in the steps are required for BAM, the procedure and syntax are exactly the same.

13.2 Changing the Location of Node Manager Log

Oracle 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).

To change the location of the Node Manager log, edit the nodemanager.properties file located in the following directory:

```
MW_HOME/wlserver_10.3/common/nodemanager
```

Oracle recommends locating this file outside of the *MW_HOME* directory, and inside the admin directory for the deployment.

Add the following line to nodemanager.properties:

```
LogFile=ORACLE_BASE/admin/nodemanager.log
```

Restart Node Manager for the change to take effect.

13.3 Enabling Host Name Verification Certificates for Node Manager in SOAHOST1

Host name verification enables communication between Node Manager and the Administration Server. This verification requires the use of certificates for the different addresses communicating with the Administration Server.

This section contains the following topics:

- Step 1: Generating Self-Signed Certificates Using the utils.CertGen Utility
- Step 2: Creating an Identity Keystore Using the utils.ImportPrivateKey Utility
- Step 3: Creating a Trust Keystore Using the Keytool Utility
- Step 4: Configuring Node Manager to Use the Custom Keystores
- Step 5: Using a Common or Shared Storage Installation

13.3.1 Generating Self-Signed Certificates Using the utils.CertGen Utility

This section describes the procedure for creating self-signed certificates on SOAHOST1.mycompany.com. Create these certificates using the network name/alias.

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.

For information on using trust CA certificates instead, see "Configuring Identity and Trust" in *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

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.

To create self-signed certificates:

1. Set up your environment by running the `WL_HOME/server/bin/setWLSEnv.sh` script:

   In the Bourne shell, run the following command:

   ```
   . setWLSEnv.sh
   ```

   Verify that the CLASSPATH environment variable is set:

   ```
   echo $CLASSPATH
   ```

2. Create a user-defined directory for the certificates:

```
mkdir certs
```

3. Change directory to the user-defined directory.

```
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:

```
java utils.CertGen password SOAHOST1.mycompany.com_cert
SOAHOST1.mycompany.com_key domestic SOAHOST1.mycompany.com
```

```
java utils.CertGen password ADMINVHN.mycompany.com_cert ADMINVHN.mycompany.com_
key domestic ADMINVHN.mycompany.com
```

13.3.2 Creating an Identity Keystore Using the utils.ImportPrivateKey Utility

This section describes how to create an Identity Keystore on SOAHOST1.mycompany.com.

In previous sections you have 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 that you use a different alias for each of the certificate/key pair imported.

1. Create a new identity keystore called appIdentityKeyStore using the utils.ImportPrivateKey utility.

   Create this keystore under the same directory as the certificates:

   ASERVER_HOME/certs

   Note: The Identity Store is created (if none exists) when you import a certificate and the corresponding key into the Identity Store using the `utils.ImportPrivateKey` utility.

   ---

2. Import the certificate and private key for both 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:

```
java utils.ImportPrivateKey appIdentityKeyStore.jks password
          appIdentity1 password
          ASERVER_HOME/certs/SOAHOST1.mycompany.com
```

```
_cert.pem
            ASERVER_HOME/certs/SOAHOST1.mycompany.com
_key.pem

java utils.ImportPrivateKey appIdentityKeyStore.jks password
            appIdentity2 password
            ASERVER_HOME/certs/SOAHOST1VHN1.mycompany
.com_cert.pem
            ASERVER_HOME/certs/SOAHOST1VHN1.mycompany
.com_key.pem

java utils.ImportPrivateKey appIdentityKeyStore.jks password
            appIdentity3 password
            ASERVER_HOME/certs/ADMINVHN.mycompany.com
_cert.pem
            ASERVER_HOME/certs/ADMINVHN.mycompany.com
_key.pem
```

13.3.3 Creating a Trust Keystore Using the Keytool Utility

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
   ASERVER_HOME/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 password -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
   password
   ```

13.3.4 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's listen address. For example on SOAHOST1, use appIdentity1 according to the steps in Section 13.3.3, "Creating a Trust Keystore Using the Keytool Utility."

```
(appIdentity1 mapped to the SOAHOST1 listen address).
Example for Node 1:
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityKeyStoreFileName=ASERVER_HOME/certs/appIdentityKeyStore.jks
CustomIdentityKeyStorePassPhrase=password
CustomIdentityAlias=appIdentity1
CustomIdentityPrivateKeyPassPhrase=password
```

The passphrase entries in the nodemanager.properties file are encrypted when you start Node Manager as described in Section 13.4, "Starting the Node Manager on SOAHOST1." For security reasons, minimize the time the entries in the nodemanager.properties file are left unencrypted. After you edit the file, start Node Manager as soon as possible so that the entries are encrypted.

13.3.5 Using a Common or Shared Storage Installation

When using a common or shared storage installation for MW_HOME, Node Manager is started from different nodes using the same base configuration (nodemanager.properties). Add the certificate for all the nodes that share the binaries to the appIdentityKeyStore.jks identity store.by creating the certificate for the new node and import it to appIdentityKeyStore.jks, as described in Section 13.3.1, "Generating Self-Signed Certificates Using the utils.CertGen Utility.". Once the certificates are available in the store, each node manager must point to a different identity alias to send the correct certificate to the Administration Server.

To set different environment variables before starting Node Manager in the different nodes:

```
cd WL_HOME/server/bin
export JAVA_OPTIONS=-DCustomIdentityAlias=appIdentitySOAHOST1

cd WL_HOME/server/bin
export JAVA_OPTIONS=-DCustomIdentityAlias=appIdentitySOAHOSTn
```

13.4 Starting the Node Manager on SOAHOST1

Start Node Manager on SOAHOST1 using the startNodeManager.sh script.

> **Note:** If you have not configured and started Node Manager yet, run the *setNMProps.sh* script as specified in section Section 8.4.2, "Starting Node Manager on SOAHOST1." After running this script you can use the start script, which is required for SOA and BAM.

To start Node Manager on SOAHOST1:

```
SOAHOST1> cd WL_HOME/server/bin
SOAHOST1> export JAVA_OPTIONS=-DCustomIdentityAlias=appIdentityX
SOAHOST1> ./startNodeManager.sh
```

> **Note:** Ensure that you specify the custom identity alias specifically assigned to each host, so `appIdentity1` for ...HOST1 and `appIdentity2` for ...HOST2, and so on.

13.5 Enabling Host Name Verification Certificates for the Node Manager in SOAHOST2

Host name verification enables communication between Node Manager and the Administration Server. This verification requires the use of certificates for the different addresses communicating with the Administration Server.

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

13.5.1 Generating Self-Signed Certificates Using the utils.CertGen Utility

This section describes the procedure for creating self-signed certificates on SOAHOST2.mycompany.com. Create these certificates using the network name/alias.

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 nodes can access the appropriate certificates. 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.

Create self-signed certificates using the untils.CertGen utility using the network name/alias.

For information on using trust CA certificates instead, see "Configuring Identity and Trust" in *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

To create self-signed certificates on SOAHOST2.mycompany.com:

1. Set up your environment by running the `WL_HOME/server/bin/setWLSEnv.sh` script:

   In the Bourne shell, run the following command:

   ```
   . setWLSEnv.sh
   ```

   Verify that the CLASSPATH environment variable is set:

   ```
   echo $CLASSPATH
   ```

2. Create a user-defined directory for the certificates:

```
mkdir certs
```

3. Change directory to the user-defined directory.

```
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:

```
java utils.CertGen password SOAHOST2_cert SOAHOST2_key
            domestic SOAHOST2.mycompany.com

java utils.CertGen password SOAHOST2VHN1_cert SOAHOST2VHN1_key
            domestic SOAHOST2VHN1.mycompany.com
```

13.5.2 Importing Identities in SOAHOST2 using the "utils.ImportPrivateKey" Utility

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.

To create an Identity Keystore on SOAHOST2.mycompany.com:

Syntax:

```
java utils.ImportPrivateKey keystore_file keystore_password certificate_alias_to_
use private_key_passphrase certificate_file private_key_file keystore_type
```

Examples:

```
java utils.ImportPrivateKey appIdentityKeyStore.jks password
            appIdentity4 password
            ORACLE_BASE/admin/domain_name/aserver/domain_name/certs/SOAHOST2_
cert.pem
            ORACLE_BASE/admin/domain_name/aserver/domain_name/certs/SOAHOST2_
key.pem

java utils.ImportPrivateKey appIdentityKeyStore.jks password
            appIdentity5 password
            ORACLE_BASE/admin/domain_name/aserver/domain_name/certs/SOAHOST2VHN1_
cert.pem
            ORACLE_BASE/admin/domain_name/aserver/domain_name/certs/SOAHOST2VHN1_
key.pem
```

13.5.3 Configuring Node Manager to Use the Custom Keystores

To configure 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=ASERVER_HOME/domain_
name/certs/appIdentityKeyStore.jks
CustomIdentityKeyStorePassPhrase=password
CustomIdentityAlias=appIdentity3
CustomIdentityPrivateKeyPassPhrase=password
```

> **Note:** The passphrase entries in the `nodemanager.properties` file get encrypted when you start Node Manager, as described in Section 13.6, "Starting 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.

13.6 Starting Node Manager on SOAHOST2

To start Node Manager on *SOAHOST2* using the startNodeManager.sh script, follow the steps in Section 13.4, "Starting the Node Manager on SOAHOST1." Run the commands from *SOAHOST2*.

> **Note:** Verify that Node Manager is using the appropriate stores and alias from the NodeManager output. Node Manager should prompt out the following:
>
> ```
> CustomIdentityKeyStoreFileName=ORACLE_BASE/admin/domain_
> name/aserver/domain_name/certs/appIdentityKeyStore.jks
> CustomIdentityAlias=appIdentityX
> ```
>
> Host name verification works if you apply a test configuration change to the servers and it succeeds without Node Manager reporting any SSL errors.

13.7 Configuring WebLogic Servers to Use the Custom Keystores

Configure the WebLogic Servers to use the custom keystores using the Oracle WebLogic Server Administration Console. Complete this procedure for the Administration Server, the WLS_WSM*n* and the WLS_SOA*n* servers.

The example directory path given in Step 6 is just an example. Oracle does not recommend putting keystores into the `aserver` directory, but recommends putting the keystore in shared storage. Having a separate directory for certificates is a better solution.

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, click **Change**, and select the "**Custom Identity and Custom Trust**" method for storing and managing private keys/digital certificate pairs and trusted CA certificates, and click **Save**.

6. In the **Identity** section, define attributes for the identity keystore.

    a. **Custom Identity Keystore**: Enter the fully qualified path to the identity keystore:

        ASERVER_HOME/certs/appIdentityKeyStore.jks

    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 13.3.2, "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:

        ASERVER_HOME/certs/appIdentityKeyStore.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 13.3.3, "Creating a Trust Keystore Using the Keytool Utility."

        As mentioned in the previous step, this attribute may be optional or required depending on the type of keystore.

8. Click **Save**.

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

10. Click **Lock & Edit**.

11. Select **Configuration**, then **SSL**.

12. In the **Private Key Alias** field, enter the alias you used for the host name the managed server listens on.

    In the **Private Key Passphrase** and the **Confirm Private Key Passphrase** fields, enter the password for the keystore that you created in Section 13.3.2, "Creating an Identity Keystore Using the utils.ImportPrivateKey Utility."

13. Click **Save**.

14. Click **Activate Changes** in the Administration Console's Change Center to make the changes take effect.

15. Restart the server for which the changes have been applied.

16. Verify that the communication between Node Manager, Administration Server and the managed servers is correct by enabling hostname verification:

    1. For each server, in the Administration Console, select **Configuration**, **SSL**, **Advanced**, **Hostname Verification**, and then **BEA HostName Verifier**.

    2. Restart the servers using the Administration Console.

14

Configuring Server Migration for an Enterprise Deployment

This chapter describes the procedures for configuring server migration for the enterprise deployment.

This chapter contains the following sections:

- Section 14.1, "Overview of Server Migration for an Enterprise Deployment"
- Section 14.2, "Setting Up a User and Tablespace for the Server Migration Leasing Table"
- Section 14.3, "Creating a GridLink Data Source for Leasing Using the Administration Console"
- Section 14.4, "Editing the Node Manager's Properties File"
- Section 14.5, "Setting Environment and Superuser Privileges for the wlsifconfig.sh Script"
- Section 14.6, "Configuring Server Migration Targets"
- Section 14.7, "Testing Server Migration"

14.1 Overview of Server Migration for an Enterprise Deployment

Configure server migration for the *WLS_SOA1* and *WLS_SOA2* managed servers. With server migration configured, should failure occur, the *WLS_SOA1* managed server restarts on *SOAHOST2*, and the *WLS_SOA2* managed server restarts on *SOAHOST1*. The *WLS_SOA1* and *WLS_SOA2* servers listen on specific floating IPs that are failed over by Oracle WebLogic Server.

Perform the steps in the following sections to configure server migration for the managed servers.

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

Set up a user and tablespace for the server migration Leasing table using the create tablespace leasing command.

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 password;

   SQL> grant create table to Leasing;

   SQL> grant create session to Leasing;

   SQL> alter user Leasing default tablespace Leasing;

   SQL> alter user Leasing quota unlimited on LEASING;
   ```

3. Create the Leasing table using the `leasing.ddl` script.

   a. Copy the `leasing.ddl` file located in either of the following directories to your database node:

   ```
   WL_HOME/server/db/oracle/817
   WL_HOME/server/db/oracle/920
   ```

   b. Connect to the database as the `Leasing` user.

   c. Run the `leasing.ddl` script in SQL*Plus.

   ```
   SQL> @copy_location/leasing.ddl;
   ```

14.3 Creating a GridLink Data Source for Leasing Using the Administration Console

Create a GridLink data source for the Leasing table from the Oracle WebLogic Server Administration Console.

To create a GridLink data source:

1. Log in to the Oracle WebLogic Server Administration Console.

2. If you have not already done so, in the **Change Center**, click **Lock & Edit** and click **Next**.

3. In the **Domain Structure** tree, expand **Services**, then select **Data Sources**.

4. On the Summary of Data Sources page, click **New** and select **GridLink Data Source**, and enter the following:

   - Enter a logical name for the data source in the **Name** field. For example, **Leasing**.

   - Enter a name for **JNDI**. For example, **jdbc/leasing**.

   - For the Database Driver, select **For the Database Driver, select Oracle's Driver (Thin) for GridLink Connections Versions: 11 and later**.

   - Click **Next**.

5. In the Transaction Options page, de-select **Supports Global Transactions**, and click **Next**.

6. In the GridLink Data Source Connection Properties Options screen, select **Enter individual listener information** and click **Next**.

7. Enter the following connection properties:

- **Service Name**: Enter the service name of the database with lowercase characters. For a GridLink data source, you must enter the Oracle RAC service name. For example:

  ```
  soaedg.mycompany.com
  ```

- **Host Name and Port**: Enter the SCAN address and port for the RAC database being used. You can identify this address by querying the appropriate parameter in the database using the TCP Protocol:

  ```
  SQL>show parameter remote_listener;

  NAME                    TYPE        VALUE

  --------------------------------------------------

  remote_listener    string      db-scan.mycompany.com
  ```

  > **Note:** For Oracle Database 11*g* Release 1 (11.1), use the virtual IP and port of each database instance listener, for example:
  >
  > ```
  > custdbhost1-vip.mycompany.com (port 1521)
  > ```
  > and
  > ```
  > custdbhost2-vip.mycompany.com (1521)
  > ```
  >
  > For Oracle Database 10*g*, use multi data sources to connect to an Oracle RAC database. For information about configuring multi data sources see Appendix A, "Using Multi Data Sources with Oracle RAC."

- **Port** - The port on which the database server listens for connection requests.

- **Database User Name**: Leasing

- **Password**: For example: welcome1

- **Confirm Password**: Enter the password again and click **Next**.

8. On the Test GridLink Database Connection page, review the connection parameters and click **Test All Listeners**. Here is an example of a successful connection notification:

   ```
   Connection test for jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_
   LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=db-scan.mycompany.com)
   (PORT=1521)))(CONNECT_DATA=(SERVICE_NAME=ps5soaedg.mycompany.com))) succeeded.
   ```

   Click **Next**.

9. In the ONS Client Configuration page, do the following:

   - Select **FAN Enabled** to subscribe to and process Oracle FAN events.

   - Enter here also the SCAN address for the RAC database and the ONS remote port as reported by the database (example below) and click **ADD**:

     ```
     [orcl@db-scan1 ~]$ srvctl config nodeapps -s

     ONS exists: Local port 6100, remote port 6200, EM port 2016
     ```

   - Click **Next**.

> **Note:** For Oracle Database 11*g* Release 1 (11.1), use the hostname and port of each database's ONS service, for example:
>
> ```
> custdbhost1.mycompany.com (port 6200)
> ```
>
> and
>
> ```
> custdbhost2.mycompany.com (6200)
> ```

10. On the Test ONS Client Configuration page, review the connection parameters and click **Test All ONS Nodes**.

    Here is an example of a successful connection notification:

    ```
    Connection test for db-scan.mycompany.com:6200 succeeded.
    ```

    Click **Next**.

11. In the Select Targets page, select **SOA_Cluster** as the target, and **All Servers in the cluster**.

12. Click **Finish**.

13. Click **Activate Changes**.

14.4 Editing the Node Manager's Properties File

Edit the Node Manager properties file on the two nodes where the servers are running. The `nodemanager.properties` file is located in the following directory:

WL_HOME/common/nodemanager

Add the following properties to enable server migration to work properly:

■   `Interface`

    ```
    Interface=eth0
    ```

    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`

    ```
    NetMask=255.255.255.0
    ```

    This property specifies the net mask for the interface for the floating IP.

■   `UseMACBroadcast`

    ```
    UseMACBroadcast=true
    ```

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 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 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.

14.5 Setting Environment and Superuser Privileges for the wlsifconfig.sh Script

Set the environment and superuser privileges for the `wlsifconfig.sh` script.

Ensure that the PATH environment variable includes the files listed in Table 14–1.

Table 14–1   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.domain	*WL_HOME*/common/nodemanager

Grant sudo privilege to the WebLogic user ('oracle') with no password restriction, and grant execute privilege on the */sbin/ifconfig* and */sbin/arping* binaries.

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.

> **Note:** Ask the system administrator for the sudo and system rights as appropriate to this step.

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`.

To grant sudo privilege to the WebLogic user ('oracle') with no password restriction, and grant execute privilege on the */sbin/ifconfig* and */sbin/arping* binaries:

```
Defaults:oracle !requiretty
oracle ALL=NOPASSWD: /sbin/ifconfig,/sbin/arping
```

14.6 Configuring Server Migration Targets

Configure server migration targets. Configuring Cluster Migration sets the `DataSourceForAutomaticMigration` property to `true`.

To configure 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 & 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:

    a. Click **Lock and Edit**

    b. In Domain Structure window of the Oracle WebLogic Server Administration Console, expand **Environment** and select **Servers**.

    c. Select the server for which you want to configure migration.

    d. Click the **Migration** tab.

e. 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**.

f. Select **Automatic Server Migration Enabled** and click **Save**.

This enables the Node Manager to start a failed server on the target node automatically.

For information on targeting applications and resources, see Appendix B, "Targeting Applications and Resources to Servers."

g. Click **Activate Changes**.

h. Restart the Administration Server and the servers for which server migration has been configured

To restart the Administration Server, use the procedure in Section 8.4.3, "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 is different from the configuration if the server is migrated automatically.

14.7 Testing Server Migration

To verify that Server Migration is working properly:

To test from Node 1:

1. Stop the WLS_SOA1 managed server.

   ```
   kill -9 pid
   ```

   pid specifies the process ID of the managed server. You can identify the pid in the node by running this command:

   ```
   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.

To test from Node 2:

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

You can also verify migration using 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 14–1   Migration Status Screen in the Administration Console
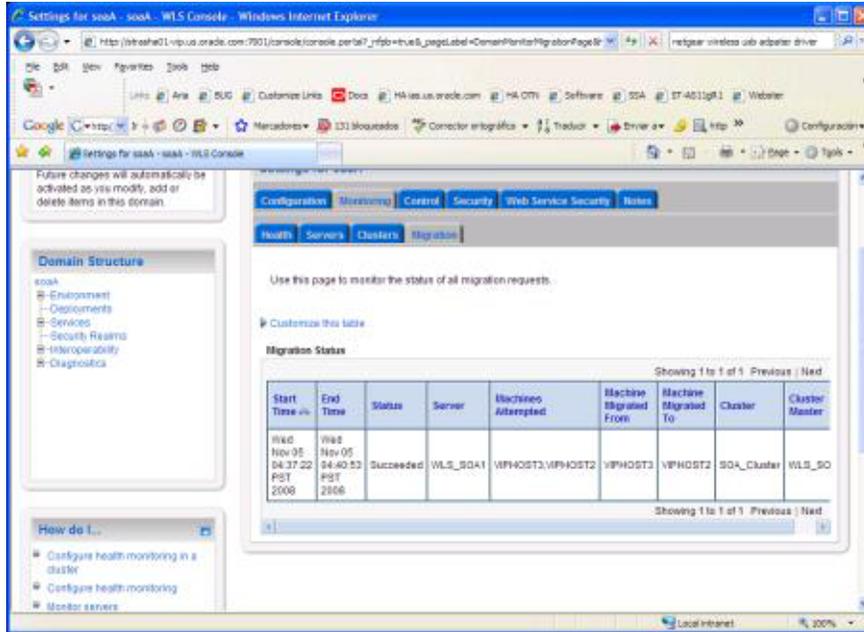
> **Note:**   After a server is migrated, to fail it back to its original node/machine, stop the managed server from the Oracle WebLogic Administration Console and then start it again. The appropriate Node Manager starts the managed server on the machine to which it was originally assigned.

15

Integrating an Enterprise Deployment with Oracle Identity Management

This chapter describes how to integrate Oracle SOA Suite with Oracle Identity Management. It contains the following sections:

- Section 15.1, "Overview of Integration With Oracle Identity Management"
- Section 15.2, "Configuring the Credential Store"
- Section 15.3, "Configuring the Policy Store"
- Section 15.4, "Re-associating Credentials and Policies"
- Section 15.5, "Oracle Access Manager 10g Integration"
- Section 15.6, "Oracle Access Manager 11g Integration"
- Section 15.7, "Backing Up the Identity Management Configuration"

15.1 Overview of Integration With Oracle Identity Management

You can integrate an Oracle Fusion Middleware enterprise deployment with Oracle Identity Manager 10*g* or 11*g*. The following sections describe how to first configure Credential and Policy stores, re-associate those credential add policy stores, and then integrate with either Oracle Identity Manager 10*g* or 11*g*.

Table 15–1 lists the high-level steps for integrating Oracle Identity Manager 10*g* with an Oracle SOA enterprise deployment.

Table 15–2 lists the high-level steps for integrating Oracle Identity Manager 11*g* with an Oracle SOA enterprise deployment.

> **Note:** When integrating with Oracle Identity Management, use the transport mode currently in use by the Oracle Identity Management servers. For example, Open, Simple or Cert.

Table 15–1   Steps for Integrating with Oracle Identity Manager 10g

Step	Description	More Information
Configure the Credential Store	Configure Oracle Internet Directory LDAP as a credential store for the Oracle Fusion Middleware SOA Suite Enterprise Deployment topology.	Section 15.2, "Configuring the Credential Store"
Configure the Policy Store	configure Oracle Internet Directory LDAP as the policy store for the Oracle Fusion Middleware SOA Suite Enterprise Deployment topology.	Section 15.3, "Configuring the Policy Store"
Use the OAM Configuration Tool	The OAM Configuration Tool (oamcfg) starts a series of scripts and setup the required policies.	Section 15.5.3, "Using the OAM Configuration Tool"
Install and Configure WebGate	Install WebGate on each of the WEBHOSTn machines in order to secure the Web tier.	Section 15.5.4, "Installing and Configuring WebGate"
Change the CacheControl Headers in the SOA_EDG_AG for Oracle BAM	Change the CacheControl headers settings in the SOA_EDG_AG Access Gate.	Section 15.5.5, "Changing the CacheControl Headers in the SOA_EDG_AG for Oracle BAM"
Configure IP Validation for the Webgate	Configure the IP validation for the Webgate using Access System Console.	Section 15.5.6, "Configuring IP Validation for the Webgate"
Set Up WebLogic Authenticators	Set up the WebLogic authenticators by backing up the configuration files, setting up the OAM ID Asserter, and Setting the order of providers.	Section 15.5.7, "Setting Up WebLogic Authenticators"

Table 15–2   Steps for Integrating with Oracle Identity Manager 11g

Step	Description	More Information
Configure the Credential Store	Configure Oracle Internet Directory LDAP as a credential store for the Oracle Fusion Middleware SOA Suite Enterprise Deployment topology.	Section 15.2, "Configuring the Credential Store"
Configure the Policy Store	configure Oracle Internet Directory LDAP as the policy store for the Oracle Fusion Middleware SOA Suite Enterprise Deployment topology.	Section 15.3, "Configuring the Policy Store"
Install WebGate	Install WebGate on each of the WEBHOST machines where an HTTP Server has already been installed.	Section 15.6.3, "Installing WebGate"

Table 15–2 (Cont.) Steps for Integrating with Oracle Identity Manager 11g

Step	Description	More Information
Register the WebGate Agent	Register the Webgate agent using the RREG tool.	Section 15.6.4, "Registering the WebGate Agent"
Set Role Members for BAMWorkflowAdmin 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 BAMWorkflowAdmin.	Section 15.6.5, "Setting Role Members for BPMWorkflowAdmin Application Role in soa-infra"
Set Up WebLogic Authenticators	Set up the WebLogic authenticators by backing up the configuration files, setting up the OAM ID Asserter, and Setting the order of providers.	Section 15.6.6, "Setting Up the WebLogic Authenticators"

15.2 Configuring the Credential Store

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 15.4, "Re-associating Credentials and Policies."

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 15.2.1, "Creating the LDAP Authenticator"
- Section 15.2.2, "Moving the WebLogic Administrator to LDAP"
- Section 15.2.3, "Reassociating the Domain Credential Store"

15.2.1 Creating the LDAP Authenticator

This section describes how to create the LDAP authenticator using the WebLogic Server Administration Console.

Prerequisites

Before you create the LDAP authenticator, 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
```

Back up the `boot.properties` file for the Administration Server in the following directory:

```
ORACLE_BASE/admin/domain_name/aserver/domain_name/servers/AdminServer/security
```

To configure the credential store to use LDAP:

1. Log in to the WebLogic Server Console.

2. Click the **Security Realms** link on the left navigational bar.

3. Click the **myrealm** default realm entry to configure it.

4. Open the **Providers** tab within the realm.

5. Observe that there is a `DefaultAuthenticator` provider configured for the realm.

6. Click **Lock & Edit**.

7. Click the **New** button to add a new provider.

8. Enter a name for the provider such as **OIDAuthenticator** or **OVDAuthenticator** depending on whether Oracle Internet Directory or Oracle Virtual Directory will be used.

9. Select the **OracleInternetDirectoryAuthenticator** or **OracleVirtualDirectoryAuthenticator** type from the list of authenticators depending on whether Oracle Internet Directory or Oracle Virtual Directory will be used and click **OK**.

10. In the Providers screen, click the newly created Authenticator.

11. Set the control flag to **SUFFICIENT**. This indicates that if a user can be authenticated successfully by this authenticator, then it should accept that authentication and should not continue to invoke any additional authenticators. If the authentication fails, it will fall through to the next authenticator in the chain. Make sure all subsequent authenticators also have their control flag set to **SUFFICIENT**; in particular, check the DefaultAuthenticator and set that to **SUFFICIENT**.

12. Click **Save** to save this setting.

13. Open the **Provider Specific** tab to enter the details for the LDAP server.

14. Enter the details specific to your LDAP server, as shown in the following table:

Parameter	Value	Value Description
Host	For example: oid.mycompany.com	The LDAP server's server ID.

Parameter	Value	Value Description
Port	For example: `636`	The LDAP server's port number.
Principal	For example: `cn=orcladmin`	The LDAP user DN used to connect to the LDAP server.
Credential	NA	The password used to connect to the LDAP server
SSL Enabled	Checked	Specifies whether SSL protocol is used when connecting to LDAP server.
User Base DN	For example: `cn=users,dc=us,dc=mycompany,dc=com`	Specify the DN under which your Users start.
Group Base DN	For example: `cn=groups,dc=us,dc=mycompany,dc=com`	Specify the DN that points to your Groups node.
Use Retrieved User Name as Principal	Checked	Must be turned on.

Click **Save** when done.

15. Click **Activate Changes** to propagate the changes.

Reorder Authenticator

Reorder the OID/OVD Authenticator and Default Authenticator and ensure that the control flag for each authenticator is set in the following order:

To set the order of the Authenticators:

1. Log in to Weblogic Console, if not already logged in.

2. Click **Lock & Edit.**

3. Navigate to **SecurityRealms**, then the default realm name, and then **Providers**.

4. Reorder the OID/OVD Authenticator, and Default Authenticator by ensuring that the control flag for each authenticator is set as follows:

   ■ OID LDAP Authenticator (or OVD LDAP Authenticator): SUFFICIENT

   ■ Default Authenticator: SUFFICIENT

5. Click **OK**.

6. Click **Activate Changes** to propagate the changes.

7. Restart the Administration Server and all managed servers.

15.2.2 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 15.2.2.1, "Provisioning Admin Users and Groups in an LDAP Directory"

■ Section 15.2.2.2, "Assigning the Admin Role to the Admin Group"

■ Section 15.2.2.3, "Updating the boot.properties File and Restarting the System"

15.2.2.1 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. Oracle does not recommend this. To avoid one WebLogic admin user having access to all the domains, the users and groups provisioned must have a unique distinguished name within the directory tree. For the SOA enterprise deployment WebLogic domain described in this guide, the admin user and group are 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
  ```

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: <password>
   mail: weblogic_soa
   objectclass: top
   objectclass: person
   objectclass: organizationalPerson
   objectclass: inetorgperson
   objectclass: orcluser
   objectclass: orcluserV2
   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 <password> -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 <password> -c -v -f admin_group.ldif
```

15.2.2.2 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.

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 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.

15.2.2.3 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:

   ```
   cd ORACLE_BASE/admin/domainName/aserver/domainName/servers/
   AdminServer/security
   ```

2. Rename the existing `boot.properties` file:

   ```
   mv boot.properties boot.properties.backup
   ```

3. Use a text editor to create a file called `boot.properties` under the security
   directory. Enter the following lines in the file:

   ```
   username=weblogic_soa
   password=password
   ```

4. Save the file.

5. Stop the Administration Server using the following command:

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

6. Start the Administrator Server using the procedure in Section 8.4.3, "Starting the
   Administration Server on SOAHOST1."

15.2.3 Reassociating the Domain Credential Store

You will complete the re-association of both the Credential and the Policy stores after
Configuring them. Re-associate the Credential and Policy stores using Enterprise
Manager Fusion Middleware Control or the WLST command
`reassociateSecurityStore`. See Section 15.4, "Re-associating Credentials and
Policies" for detailed steps.

15.3 Configuring the Policy Store

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. This procedure consists of two parts:

- Setting a Node in The Server Directory
- Re-associating the Domain Policy Store

For more information on policy store configuration, see "OPSS Authorization and the Policy Store" chapter in the *Oracle Fusion Middleware Security Guide*.

15.3.1 Setting a Node in The Server Directory

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. These steps should be completed by an Oracle Internet Directory administrator.

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 SQL*Plus, 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, see the *Oracle Fusion Middleware User Reference for Oracle Identity Management*.

15.3.2 Re-associating the Domain Policy Store

Reassociate the policy store by migrating policy data from a file- or LDAP-based repository to an LDAP-based repository. 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, updates the matching policy as appropriate. If none are found, it 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.

15.4 Re-associating Credentials and Policies

This section describes the procedure for re-associate the policy and credential store with Oracle Internet Directory using the WLST `reassociateSecurityStore` command.

To re-associate the policy and credential stores:

1. From SOAHOST1, start the `wlst` shell:

```
cd ORACLE_COMMONHOME/common/bin
./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","password","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="password",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=password}
Location changed to domainRuntime tree. This is a read-only tree with
DomainMBean as the root.

For more help, use help(domainRuntime)

Starting Policy Store reassociation.
LDAP server and  ServiceConfigurator setup done.
```

```
Schema is seeded into LDAP server
Data is migrated to LDAP server
Service in 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 8.4.3, "Starting the Administration Server on SOAHOST1."

> **Note:** For credential and policy changes to take effect, the servers in the domain must be restarted.

Cataloging Oracle Internet Directory Attributes

Index any Oracle Internet Directory attribute that is used in a search filter. The indexing is an optional procedure that enhances performance. If you have not yet indexed the attributes in this Oracle Internet Directory, use the `catalog` tool to index them.

For example, to index the `orclerolescope` attribute:

```
catalog connect="orcl" add=true attribute="orclrolescope" verbose="true"
```

You can also index multiple attribute names by listing them in a file and processing them as a batch as follows:

```
orclrolescope
orclassignedroles
orclApplicationCommonName
orclAppFullName
orclCSFAlias
orclCSFKey
orclCSFName
orclCSFDBUrl
orclCSFDBPort
orclCSFCredentialType
orclCSFExpiryTime
modifytimestamp
createtimestamp
orcljpsassignee
```

For more information about indexing OID attributes, see Tasks and Examples for catalog in the *Oracle Fusion Middleware Reference for Oracle Identity Management*.

15.5 Oracle Access Manager 10*g* Integration

This section describes how to set up Oracle Access Manager 10*g* as the single sign-on solution for the Oracle SOA Suite Enterprise Deployment topology.

This section contains the following Topics:

- Section 15.5.1, "Overview of Oracle Access Manager Integration"
- Section 15.5.2, "Prerequisites for Oracle Access Manager"
- Section 15.5.3, "Using the OAM Configuration Tool"
- Section 15.5.4, "Installing and Configuring WebGate"
- Section 15.5.5, "Changing the CacheControl Headers in the SOA_EDG_AG for Oracle BAM"
- Section 15.5.6, "Configuring IP Validation for the Webgate"
- Section 15.5.7, "Setting Up WebLogic Authenticators"

15.5.1 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*.

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 10*g* environment requires additional configuration on the WebGate to handle the end_url. Without this additional configuration, you are logged out, but not redirected to the end URL because Oracle Access Manager 10*g* WebGate does not process end_url.

For information about configuration procedures, see *Oracle Fusion Middleware Security Guide*.

15.5.2 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.

15.5.3 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 15.5.3.1, "Prerequisites for Running the OAM Configuration Tool"
- Section 15.5.3.2, "Running the OAM Configuration Tool"
- Section 15.5.3.3, "Verifying Successful Creation of the Policy Domain and AccessGate"
- Section 15.5.3.4, "Updating the Host Identifier"
- Section 15.5.3.5, "Updating the WebGate Profile"
- Section 15.5.3.6, "Adding Additional Access Servers"
- Section 15.5.3.7, "Configuring Delegated Form Authentication"

15.5.3.1 Prerequisites for Running the OAM Configuration Tool

Review the following prerequisites before running the OAM Configuration Tool:

- **Password:** Create a secure password. This will be used as the password for the WebGate installation created later.
- **LDAP Host:** Have the host name of the Directory Server or Load Balancer address available in the case of a high availability or enterprise deployment configuration.
- **LDAP Port:** Have the port of the Directory Server available.
- **LDAP USER DN:** Have the DN of the LDAP admin user available. This is a value such as "cn=orcladmin."
- **LDAP password:** Have the password of the LDAP admin user available.
- **oam_aa_host:** Have the host name of an Oracle Access Manager available.
- **oam_aa_port:** Have the port of the Oracle Access Manager available.

15.5.3.2 Running the OAM Configuration Tool

You can find the OAM Configuration Tool at the following location:

ORACLE_COMMON_HOME/modules/oracle.oamprovider_11.1.1

ORACLE_COMMON_HOME depends on the machine on which you are running the configuration tool. The tool can be run from any machine with the required installation files. The procedure described in this section runs the tool from *SOAHOST1*.

The OAM Configuration Tool provides a way to register protected and public resources into the OAM system. The following is a list of protected resources to be added to the OAM system:

```
/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)
/BAM/composer (For BAM systems only)
/BAM/composer/.../* (For BAM systems only)
/BAM/workspace
/BAM/workspace/.../*
/soa-infra
/soa-infra/deployer
/soa-infra/events/edn-db-log
/soa-infra/cluster/info
/inspection.wsil/
/sbconsole/
/sbconsole/.../*
```

The following is a list of public resources:

```
/soa-infra/services/.../*
/soa-infra/directWSDL
/soa-infra/directWSDL/.../*
/OracleBAMWS
/OracleBAMWS/.../*
/ucs/messaging/webservice
/ucs/messaging/webservice/.../*
/sbinspection.wsil /sbinpection.wsil/.../*
/osb
/osb/.../*
/sbresource
/sbresource/.../*
/b2b/services/ws/
/b2b/services/ws/.../*
```

Where "/.../*" implies all resources under the base url context.

To run the OAM Configuration Tool:

1. Run the OAM Configuration Tool for OAM 10*g* registration on a single command line using the following command:

```
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
```

2. Define the $URI_LIST variable to contain the list of URIs you want to protect as follows:

```
##########################
#Product Name: SOA
##########################

#######################
protected_uris
#######################

/integration/worklistapp
/workflow/sdpmessagingsca-ui-worklist
/b2bconsole
/sdpmessaging/userprefs-ui
/DefaultToDoTaskFlow
/em
/console
/soa/composer
/soa-infra
/soa-infra/deployer
/soa-infra/events/edn-db-log
/soa-infra/cluster/info
/inspection.wsil
#(For BAM systems only)
/OracleBAM
# For BAM systems only)
/BAM/composer
/BAM/workspace
/sbconsole
/sbconsole/.../*
```

3. Define the $PUBLIC_URI_LIST variable to contain the list of URIs you want to set as not protected/public as follows:

```
##########################
public_uris
##########################

/soa-infra/services
/soa-infra/directWSDL
/OracleBAMWS
/ucs/messaging/webservice
/sbinspection.wsil
/osb
/b2b/services/ws/
/sbresource
```

> **Note:** In OAM 10*g* all resources under a URL prefix are protected by the default rules of a policy domain unless more specific rules are applied to them through policies. Refer to the 10*g* version of the *Oracle Access Manager Access Administration Guide* for details on the different patterns that can be used if more specialized protection patterns need to be used.

4. To validate the 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
```

> **Note:** If BAM is installed later or other additional URLs need to be protected, run the OAM Configuration Tool again using the same `app_domain` and include *all* the URLs to be protected (not just the new ones).

15.5.3.3 Verifying Successful Creation of the Policy Domain and AccessGate

There are two parts to the procedure for verifying creation of the policy domain and the AccessGate:

To verify the policy domain:

1. Log on to the Oracle Access Manager using the following URL:

   ```
   http://OAMADMINHOST:port/access/oblix/
   ```

2. Click **Policy Manager**.

3. Click the **My Policy Domains** link on the left panel.

   A list of all policy domains appears. The domain you just created will be listed there. It will have the suffix _PD (for example, SOA_EDG_PD). In the third column URL prefixes, the URIs you specified during the creation of this domain appear.

4. Click the link to the policy domain you just created.

   This link takes you to the General area of this domain.

5. Click the **Resources** tab.

   The URIs you specified appear. You can also click other tabs to view other settings.

To verify the AccessGate configuration:

1. Click the **Access System Console** link on the top right hand side.

   This acts like a toggle; after you click it, it becomes the **Policy Manager** link.

2. Click the **Access System Configuration** tab.

3. Click the **AccessGate Configuration** link on the left panel.

4. Enter **SOA_EDG** as the search criterion (or any other substring you may have used as the `app_domain` name in Section 15.5.3.2, "Running the OAM

Configuration Tool"), and click **Go**.

5. Once the AccessGate for the domain you just created appears (this will have the suffix _AG (for example, `SOA_EDG_AG`), click it, and the details of the AccessGate which you just created appear.

15.5.3.4 Updating the Host Identifier

The OAM Configuration Tool uses the value of the `app_domain` parameter to create a host identifier for the policy domain. This host identifier must be updated with all the host name variations for the host so that the configuration works correctly. Follow the steps below to update the host identifier created by the OAM Configuration Tool:

1. Navigate to the Access System Console by specifying the following URL in your web browser:

   ```
   http://hostname:port/access/oblix
   ```

   where *hostname* refers to the host where WebPass Oracle HTTP Server instance is running and *port* refers to the HTTP port of the Oracle HTTP Server instance.

2. When prompted for a username and password, log in as an administrator. Click **OK**.

3. On the Access System main page, click the **Access System Console** link.

4. On the Access System Console page, click the Access System Configuration tab.

5. On the Access System Configuration page, click **Host Identifiers** at the bottom left.

6. On the List all host identifiers page, click on the host identifier created by the OAM Configuration Tool. For example, select `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

- soahost1vhn2:8011

- soahost2vhn2:8011

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.

15.5.3.5 Updating the WebGate Profile

The OAM Configuration Tool populates the `Preferred_HTTP_Host` and hostname attributes for the WebGate profile that is created with the value of the `app_domain` parameter. Both these attributes must be updated with the proper values for the configuration to work correctly. Follow the steps below to update the WebGate profile created by the OAM CFG Tool.

1. Navigate to the Access System Console by specifying the following URL in your web browser:

   ```
   http://hostname:port/access/oblix
   ```

   where `hostname` refers to the host where WebPass Oracle HTTP Server instance is running and `port` refers to the HTTP port of the Oracle HTTP Server instance.

2. On the Access System main page, click the **Access System Console** link, then log in as an administrator.

3. On the Access System Console main page, click **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.

15.5.3.6 Adding Additional Access Servers

To assign an Access Server to the WebGate:

1. Log in as the Administrator on the Access System Console.

2. Navigate to the **Details** for AccessGate page, if necessary. From the Access System Console, select **Access System Configuration**, then **AccessGate Configuration**, then the link for the WebGate (**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.

15.5.3.7 Configuring Delegated Form Authentication

To configure the form authentication to redirect to the WebGate that was installed with the OAM installation:

1. Open the Access System Console.

2. In the Access System Configuration screen, select **Authentication Management** from the left-hand bar.

3. Select **OraDefaultFormAuthNScheme**.

4. Click **Modify**.

5. In the Challenge Redirect field, enter the host and port of the IDM installation; for example: `http://sso.mycompany.com`.

A WebGate should already be installed in the IDM installation. Refer to *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management* for details.

15.5.4 Installing and Configuring WebGate

Install WebGate on each of the WEBHOST*n* machines in order to secure the Web tier.

To install and configure WebGate:

1. Launch the WebGate installer (see Section 2.3, "Identifying the Software Components to Install" for information on where to obtain it) using the following command:

   ```
   ./Oracle_Access_Manager10_1_4_3_0_linux_OHS11g_WebGate -gui
   ```

   The Welcome screen appears. Click **Next**.

2. In the Customer Information screen (Figure 15–1), enter the user name and user group that the web server is running as. Click **Next** to continue.

Figure 15–1   Customer Information Screen

3. In the installation target screen (Figure 15–2), specify the directory where WebGate should be installed. Click **Next** to continue.

Figure 15–2   Installation Target Screen
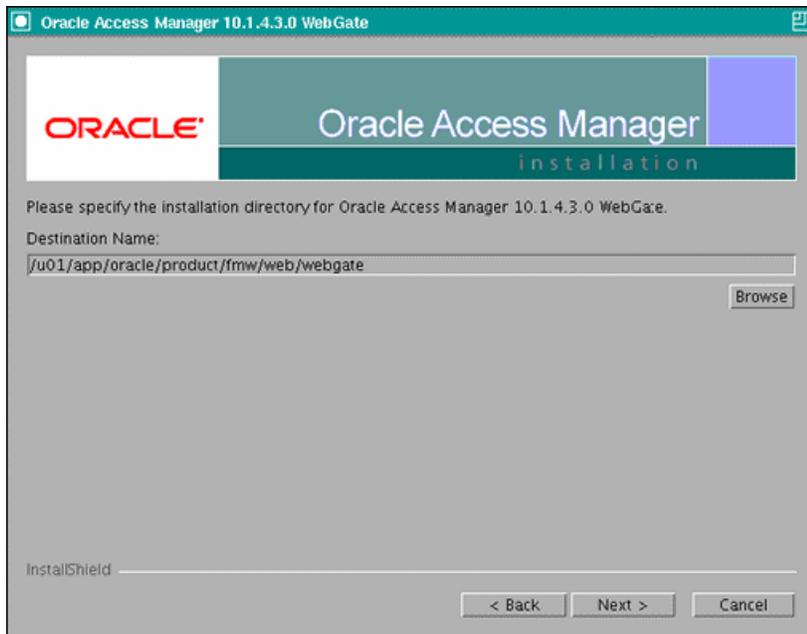
4. In the installation summary screen, click **Next**.

5. Download the required GCC runtime libraries for WebGate as instructed in the WebGate configuration screen (Figure 15–3), and use **Browse** to point to their location on the local computer. Click **Next** to continue.

Figure 15–3   Runtime Libraries Screen

6. The installer now creates the required artifacts. After that is completed, click **Next** to continue.

7. In the transport security mode screen (Figure 15–4), select "Open Mode: No Encryption" and click **Next** to continue.

Figure 15–4   Transport Security Mode Screen
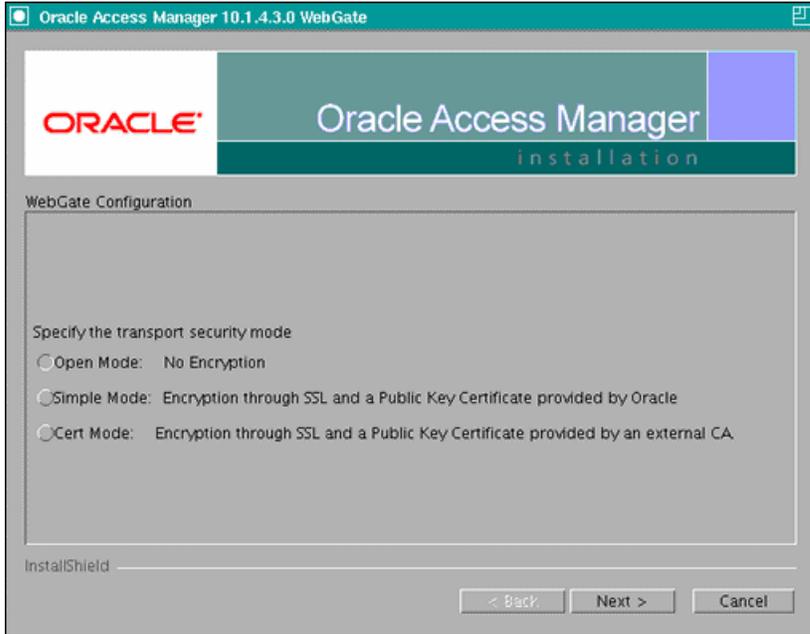
8. In the WebGate configuration screen, provide the details of the Access Server that will be used. You must provide the following information:

   ■ **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 15–5   Access Server Configuration Screen

9. In the Configure Web Server screen, click **Yes** to automatically update the web server. Click **Next** to continue.

10. 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:

    `ORACLE_BASE/admin/ohs_instance2/config/OHS/ohs2/httpd.conf`

    Click **Next** to continue.

11. 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.

12. Stop and start your Web server for the configuration updates to take effect. Click **Next** to continue.

13. 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.

14. 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.

15. 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.

16. The Oracle COREid Readme screen appears. Review the information on the screen and click **Next** to continue.

    A message appears (along with the details of the installation) informing you that the installation was successful.

15.5.5 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. Change the CacheControl headers settings in the SOA_EDG_AG Access Gate to prevent these errors.

To change the CacheControl headers 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.

15.5.6 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:

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.

15.5.7 Setting Up WebLogic Authenticators

This section describes how to set up WebLogic Authenticators.

Prerequisite

If you have not already created the LDAP authenticator, do it before continuing with this section. To set up the LDAP authenticator, follow the steps in Section 15.2.1, "Creating the LDAP Authenticator."

This section includes the following topics:

- Section 15.5.7.1, "Back Up Configuration Files"
- Section 15.5.7.2, "Setting Up the OAM ID Asserter"
- Section 15.5.7.3, "Setting the Order of Providers"

15.5.7.1 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.

15.5.7.2 Setting Up the OAM ID Asserter

Set up the OAM ID Asserter using the Weblogic Console.

To set up the OAM ID Asserter:

1. Log into Weblogic Console, if not already logged in.

2. Navigate to the following location:

   ```
   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.

15.5.7.3 Setting the Order of Providers

To set the order of the providers:

1. Log in to Weblogic Console, if not already logged in.

2. Click **Lock & Edit.**

3. Navigate to **SecurityRealms**, then the default realm name, and then **Providers**.

4. Click **Reorder**.

5. Reorder the OAM Identity Asserter, OID/OVD Authenticator, and Default Authenticator by ensuring that the control flag for each authenticator is set as follows:

   - OAM Identity Asserter: REQUIRED

   - OID LDAP Authenticator (or OVD LDAP Authenticator): SUFFICIENT

   - Default Authenticator: SUFFICIENT

6. Click **OK**.

7. Click **Activate Changes** to propagate the changes.

8. Restart the Administration Server and all managed servers.

15.6 Oracle Access Manager 11*g* Integration

This section describes how to set up Oracle Access Manager 11*g* as the single sign-on solution for the Oracle SOA Enterprise Deployment topology.

This section contains the following sections:

- Section 15.6.1, "Overview of Oracle Access Manager Integration"

- Section 15.6.2, "Prerequisites for Oracle Access Manager"

- Section 15.6.3, "Installing WebGate"

- Section 15.6.4, "Registering the WebGate Agent"

- Section 15.6.5, "Setting Role Members for BPMWorkflowAdmin Application Role in soa-infra"

- Section 15.6.6, "Setting Up the WebLogic Authenticators"

15.6.1 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 11*g* 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*.

15.6.2 Prerequisites for Oracle Access Manager

Before completing the procedures in this section you must have an existing Oracle Access Manager (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.

15.6.3 Installing WebGate

This section describes how to install WebGate on each of the WEBHOST machines where an HTTP Server has already been installed.

15.6.3.1 Prerequisite for Installing GCC Libraries

Before installing WebGate, download and install third-party GCC libraries on your machine. 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 version 3.3.2. Table 15–3 lists the versions of GCC third-party libraries for Linux and Solaris.

Table 15–3   Versions of GCC Third-Party Libraries for Linux and Solaris

Operating System	Architecture	GCC Libraries	Required Library Version
Linux 32-bit	x86	libgcc_s.so.1	3.3.2
		libstdc++.so.5	
Linux 64-bit	x64	libgcc_s.so.1	3.4.6
		libstdc++.so.6	
Solaris 64-bit	SPARC	libgcc_s.so.1	3.3.2
		libstdc++.so.5	

15.6.3.2 Installing WebGate

This section describes the procedures for installing WebGate.

The Installer program for Oracle HTTP Server 11*g* Webgate for Oracle Access Manager is included in the webgate.zip file.

To install WebGate:

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. Set the MW_HOME environment variable to the Middleware Home for the web tier:

    ```
    export MW_HOME=ORACLE_BASE/product/fmw/web
    ```

4. Start the installer using the following command:

    ```
    $ ./runInstaller -jreLoc MW_HOME/jdk
    ```

    Note: When you install Oracle HTTP Server, the jdk directory is created under the *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.

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

6. In the Prerequisite Checks screen, click **Next**.

7. In the Specify Installation Location screen, specify the Oracle Middleware Home and Oracle Home locations.

    ■ *ORACLE_BASE*/product/fmw

    ■ Oracle_OAMWebGate1 (leave the default name)

    Note: The Middleware Home contains an Oracle Home for Oracle Web Tier. The default name is Oracle_OAMWebGate1 for this Oracle home directory, which is created under the Middleware Home.

    Click **Next**.

8. In the Specify GCC Library screen, specify the directory that contains the GCC libraries, and click **Next**.

9. In the Installation Summary screen, verify the information on this screen and click **Install** to begin the installation.

10. 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.

11. In the Installation Complete screen, click **Finish** to exit the installer.

15.6.3.3 Post-Installation Steps

Complete the following procedure after installing Oracle HTTP Server 11*g* Webgate for Oracle Access Manager:

1. Move to the following directory under your Oracle Home for Webgate:

```
$ cd Webgate_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 ORACLE_BASE/admin/webN/config/OHS/ohsN
-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 following directory is the Instance Home of an Oracle HTTP Server (where N is a sequential number for your installation; for example, 1 for WEBHOST1 or 2 for WEBHOST2).

```
ORACLE_BASE/admin/webN/config/OHS/ohsN
```

> **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_Oracle_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 ORACLE_BASE/admin/webN/config/OHS/ohsN [-oh Webgate_Oracle_
Home]
[-o output_file]]
```

> **Note:** The -oh *WebGate_Oracle_Home* and -o *output_file* parameters
> are optional.

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
```

15.6.4 Registering the WebGate Agent

This section describes the procedures for registering the WebGate Agent.

15.6.4.1 The RREG Tool

The RREG tool is part of the OAM 11*g* installation. If it is not already available, extract
it using the following procedure:

1. After installing and configuring Oracle Access Manager, navigate to the following
    location:

    ```
    IDM_Home/oam/server/rreg/client
    ```

2. On the command line, untar the RREG.tar.gz file using gunzip, as in the
    following example:

    ```
    gunzip RREG.tar.gz

    tar -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/.../*
/integration/services
/integration/services/.../*
/workflow/sdpmessagingsca-ui-worklist
/workflow/sdpmessagingsca-ui-worklist/.../*
/b2bconsole
/b2bconsole/.../*
/sdpmessaging/userprefs-ui
```

```
/sdpmessaging/userprefs-ui/.../*
/workflow/DefaultToDoTaskFlow
/workflow/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 (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/
/sbconsole
/sbconsole/.../*
```

The list of public resources is:

```
/soa-infra/services/.../*
/soa-infra/directWSDL
/soa-infra/directWSDL/.../*
/OracleBAMWS
/OracleBAMWS/.../*
/b2b/services/ws/
/b2b/services/ws/.../*
/wsm-pm
/wsm-pm/.../*
```

Where "/.../*" implies all resources under the base url context.

15.6.4.2 Updating the OAM11*g*Request file

In the *RREG_Home*/input directory there are template files named
`OAM11GRequest.xml`. Copy and edit his file to create the policies for the SOA
installation.

Add execute permissions for the `oamreg.sh` script:

```
chmod u+x /RREG_Home/bin/oamreg.sh
```

After editing, the file should appear 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>
    <ValList ListName="ipValidationExceptions">
        <ValListMember Value="10.1.1.1"/>
    </ValList>
    <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>/workflow/DefaultToDoTaskFlow</resource>
        <resource>/workflow/DefaultToDoTaskFlow/.../*</resource>
        <resource>/em</resource>
        <resource>/em/.../*</resource>
        <resource>/console</resource>
        <resource>/console/.../*</resource>
        <resource>/sbconsole</resource>
        <resource>/sbconsole/.../*</resource> <-- (For OSB systems only) -->
        <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>/integration/services/.../*</resource>
        <resource>/integration/services</resource>
        <resource>/soa-infra/directWSDL</resource>
        <resource>/b2b/services/ws</resource>
        <resource>/b2b/services/ws/../* </resource>
        <resource>/sbinspection.wsil</resource> <!-- (For OSB systems only) -->
    </publicResourcesList>
```

```
    <excludedResourcesList>
        <resource>/wsm-pm</resource>
        <resource>/wsm-pm/.../*</resource>
        <resource>/soa-infra/services/.../*</resource>
        <resource>/OracleBAMWS</resource> <!-- (For BAM systems only) -->
        <resource>/OracleBAMWS/.../*</resource><!-- (For BAM systems only) -->
        <resource>/ucs/messaging/webservice</resource>
        <resource>/ucs/messaging/webservice/.../*</resource>
        <resource>/osb</resource> <!-- (For OSB systems only) -->
        <resource>/osb/.../*</resource> <!-- (For OSB systems only) -->
        <resource>/sbresource</resource> <!-- (For OSB systems only) -->
        <resource>/sbresource/.../*</resource> <!-- (For OSB systems only) -->
    </excludedResourcesList>
</OAM11GRegRequest>
```

15.6.4.3 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.
```

15.6.4.4 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 location on the WEBHOST machine:

```
scp ObAccessClient.xml oracle@WEBHOSTN:ORACLE_BASE/admin/webN/config/OHS/ohsN/
webgate/config/

scp cwallet.sso oracle@WEBHOSTN:ORACLE_BASE/admin/webN/config/OHS/ohsN/
webgate/config/
```

In the `scp` command, *N* is a sequential number for your installation; for example, 1 for WEBHOST1 or 2 for WEBHOST2.

Restart the Web tier instances to activate the WebGate.

15.6.5 Setting 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 BAMWorkflowAdmin.

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', 'SOADMINHOST: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")
   ```

15.6.6 Setting Up the WebLogic Authenticators

Set up the WebLogic authenticators by backing up the configuration files, setting up the OAM ID Asserter, and setting the order of providers.

Prerequisite

Before you set up the WebLogic authenticators, you should have already set up the LDAP authenticator by following the steps in Section 15.2.1, "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 15.6.6.1, "Back Up Configuration Files"

- Section 15.6.6.2, "Setting Up the OAM ID Asserter"

- Section 15.6.6.3, "Setting the Order of Providers"

15.6.6.1 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.

15.6.6.2 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 & 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 Chosen types.

9. Save the settings.

10. Click **Apply Changes**.

Finally, log in as admin to WLST console and run the following command:

```
cd /ORACLE_HOME/Oracle_SOA1/common/bin
./wlst.sh
addOAMSSOProvider(loginuri="/${app.context}/adfAuthentication",logouturi="/oamsso/
logout.html")
```

15.6.6.3 Setting the Order of Providers

Set the order of providers using the WebLogic Administration Console.

To set the order of the providers:

1. Log in to Weblogic Console.

2. Click **Lock & Edit.**

3. Navigate to **SecurityRealms**, then the default realm name, and then **Providers**.

4. Reorder the OAM Identity Asserter, OID/OVD Authenticator, and Default Authenticator by ensuring that the control flag for each authenticator is set as follows:

   - OAM Identity Asserter: REQUIRED

   - OID LDAP Authenticator (or OVD LDAP Authenticator): SUFFICIENT

   - Default Authenticator: SUFFICIENT

5. Click **OK**.

6. Click **Activate Changes** to propagate the changes.

7. Restart the Administration Server and all managed servers.

15.7 Backing Up the Identity Management Configuration

After you have verified that the extended domain is working, back up the domain configuration. This is a quick backup for the express purpose of immediate restore in case of failures in future procedures. Back up the configuration to the local disk. This backup can be discarded once you have completed the enterprise deployment. Once you have completed the enterprise deployment, you can initiate the regular deployment-specific backup and recovery process.

For information about backing up the environment, see "Backing Up Your Environment" in the *Oracle Fusion Middleware Administrator's Guide*. For information about recovering your information, see "Recovering Your Environment" in the *Oracle Fusion Middleware Administrator's Guide*.

To back up the configuration 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 Administration Server domain directory to save your domain configuration. The configuration files are located in the following directory:

   `ORACLE_BASE/ admin/domain_name`

   To back up the Administration Server run the following command on SOAHOST1:

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

16

Managing the Topology for an Enterprise Deployment

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:

16.1 Overview of Managing the Topology

After configuring the SOA enterprise deployment, use the information in this chapter to manage the topology.

SOA applications are deployed as composites, consisting of different kinds of components. SOA composite applications include the following:

- Service components such as Oracle Mediator for routing, BPEL processes for orchestration, BAMN processes for orchestration (if Oracle BAM Suite is also installed), human tasks for workflow approvals, spring for integrating Java interfaces into SOA composite applications, and decision services for working with business rules.

- Binding components (services and references) for connecting SOA composite applications to external services, applications, and technologies.

These components are assembled into a single SOA composite application. This chapter offers tips for deploying SOA composite applications.

For information on monitoring SOA composite applications, see Monitoring SOA Composite Applications in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suit*.

For information on managing SOA composite applications, see Managing SOA Composite Applications in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suit*.

At some point you may need to expand the topology by scaling it up, or out. See Section 16.5, "Scaling Up the Topology (Adding Managed Servers to Existing Nodes)," and Section 16.6, "Scaling Out the Topology (Adding Managed Servers to New Nodes)" for information about the difference between scaling up and scaling out, and instructions for performing these tasks.

Back up the topology before and after any configuration changes. Section 16.8, "Performing Backups and Recoveries in the SOA Enterprise Deployments" provides information about the directories and files that should be back up to protect against failure as a result of configuration changes.

This chapter also documents solutions for possible known issues that may occur after you have configured the topology.

16.2 Tips for Deploying Composites and Artifacts in a SOA Enterprise Deployment Topology

This section describes tips for deploying composites and artifacts for a SOA enterprise deployment. See the "Deploying SOA Composite Applications" chapter in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suit* for instructions on deploying composites.

Deploy composites to a specific server address

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.

Use B2B Console for deployment agreements and purge/import metadata

For B2B, deploy agreements and purge/import metadata ONLY from the GUI available in B2B console. Do not use the command line utility. Using the command line utility for these operations may cause inconsistencies and errors in the B2B system.

Additional instructions for FOD deployment

If you are deploying the SOA Fusion Order Demo, complete the deployment steps provided in the FOD's README file, and then complete the following additional steps:

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"
```

16.3 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.

To manage space in the SOA infrastructure database:

- Watch for generic errors, such as "oracle.fabric.common.FabricInvocationException" in the Oracle Enterprise Manager Fusion Middleware Control console (dashboard for instances).

- Search 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.

- Purge old composite instances 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. 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. For more information on the possible operations included in the SQL packages provided, see "Managing SOA Composite Applications" in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite*. Always use the scripts provided for a correct purge. Deleting rows in just the composite_dn table may leave dangling references in other tables used by the Oracle Fusion Middleware SOA Infrastructure.

16.4 Configuring UMS Drivers

UMS driver configuration is not automatically propagated in a SOA or BAM cluster. To propagate UMS driver configuration in a cluster:

- Apply the UMS driver configuration in each server in the Enterprise Deployment topology that is using the driver.

- If you are using server migration, servers are moved to a different node's domain directory. Pre-create the UMS driver configuration in the failover node. The UMS driver configuration file is located in the following directory:

  ORACLE_BASE/admin/*domain_name*/mserver/*domain_name*/servers/*server_name*/ tmp/_WL_
  user/*ums_driver_name*/*/configuration/driverconfig.xml

  Where '*' represents a directory name that is randomly generated by Oracle WebLogic Server during deployment. For example, 3682yq.

Create the UMS driver configuration file in preparation for possible failovers by forcing a server migration, and copy the file from the source node.

For example, to create the file for BAM:

1. Configure the driver for WLS_BAM1 in BAMHOST1.

2. Force a failover of WLS_BAM1 to BAMHOST2. Verify the following 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/
```

4. Restart the driver for these changes to take effect.

To restart the driver:

   a. Log on to the Oracle WebLogic Administration Console.

   b. Expand the environment node on the navigation tree.

   c. Click on **Deployments**.

   d. Select the driver.

   e. Click **Stop->When work completes** and confirm the operation.

   f. Wait for the driver to transition to the "Prepared" state (refresh the administration console page, if required).

   g. Select the driver again, and click **Start->Servicing all requests** and confirm the operation.

Verify in Oracle Enterprise Manager Fusion Middleware Control that the properties for the driver have been preserved.

16.5 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 Fusion Middleware 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 these existing 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.

When you scale up a server that uses server migration, plan for your appropriate capacity and resource allocation needs. Take the following scenario for example:

- Server1 exists in node1 and uses server migration in its cluster with server2 on node2.

- Server3 is added to the cluster in node1 in a scale up operation. It also uses server migration.

In this scenario, a situation may occur where all servers (server1, server2, server3 and admin server) end up running in a node1 or node2. This means each node needs to be designed with enough resources to sustain the worst case scenario where all servers using server migration end in one single node (as defined in the server migration candidate machine configuration).

> **Note:** Scale up procedure apply only to WSM managed servers.

16.5.1 Scale-up Procedure for Oracle SOA

To scale up the SOA topology:

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:

    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 & Edit** and select the managed server that you want to clone (for example, WLS_SOA1).

    c. Click **Clone**.

    d. Name the new managed server WLS_SOA*n*, where *n* is a number that identifies the new managed server. In this case, you are adding a new server to Node 1, where WLS_SOA1 was running.

    For the remainder of the steps, 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 virtual IP (also called a floating IP) to enable it to move to another node. The virtual IP 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 8.6, "Configuring the Java Object Cache for Oracle WSM." You can use the same discover port for multiple WLS_WSM servers in the same node. Repeat the steps provided in Section 8.6 for each WLS_WSM server and the server list is updated.

4. Create JMS servers for SOA and UMS on the new managed server.

    > **Note:**   You do not have to create JMS servers for SOA and UMS on the new managed server if you are scaling up the WLS_WSM managed server or the BAM Web Applications system. This procedure is required only if you are scaling up the WLS_SOA managed servers.

    To create the JMS servers for SOA and UMS:

    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 Chapter 4, "Preparing the File System for an Enterprise Deployment," as the directory for the JMS persistent stores:

    `ORACLE_BASE/admin/domain_name/cluster_name/jms`

    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 Chapter 4, "Preparing the File System for an Enterprise Deployment," as the directory for the JMS persistent stores:

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

> **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. 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*).

    f. Update the SubDeployment Targets for SOA and UMS 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 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, or UMSJMSServerXXXXXX, 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).

5. Configuring Oracle Coherence for deploying composites for the new server as described in Section 9.4, "Configuring Oracle Coherence for Deploying Composites."

> **Note:** Only the **localhost** field must be changed for the server. Replace the localhost with the listen address of the new server added:
>
> Dtangosol.coherence.localhost=SOAHOST1VHNn

6. Reconfigure the JMS Adapter with the new server using the **FactoryProperties** field in the Administration Console. Click on the corresponding cell under the **Property** value and enter the following:

```
java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory;java.naming.p
rovider.url=t3://soahostvhn1:8001,soahos2tvhn1:8001;java.naming.security.princi
pal=weblogic;java.naming.security.credentials=weblogic
```

Click **Save and Activate**.

7. Configure the persistent store for the new server. This should be a location visible from other nodes as recommended in Chapter 4, "Preparing the File System for an Enterprise Deployment."

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.

8. 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**.

9. Configure server migration for the new managed server. To configure server migration using the Oracle WebLogic Server Administration Console:

> **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.

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 and click **Save**.

This enables the Node Manager to start a failed server on the target node automatically.

f. Restart the Administration Server, managed servers, and Node Manager.

To restart the Administration Server, use the procedure in Section 8.4.3, "Starting the Administration Server on SOAHOST1.".

10. 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 & 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.

11. 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 by running the following command:

```
kill -9 pid
```

You can identify the PID (process ID) of the node using the following command:

```
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. Node Manager logs a message indicating that the server will not be restarted again locally.

16.5.2 Scale-up Procedure for Oracle BAM

To scale up the SOA topology:

1. Using the Oracle WebLogic Server Administration Console, clone WLS_SOA1 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:

   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 & Edit** and select the managed server that you want to clone (for example, WLS_SOA1).

   c. Click **Clone**.

   d. Name the new managed server WLS_SOA*n*, where *n* is a number that identifies the new managed server. In this case, you are adding a new server to Node 1, where WLS_SOA1 was running.

   For the remainder of the steps, 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 virtual IP (also called a floating IP) to enable it to move to another node. The virtual IP should be different from the one used by the managed server that is already running.

3. Create JMS servers for SOA and UMS on the new managed server.

   To create the JMS servers for SOA, UMS and BAM:

   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 Chapter 4, "Preparing the File System for an Enterprise Deployment," as the directory for the JMS persistent stores:

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

   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 Chapter 4, "Preparing the File System for an Enterprise Deployment," as the directory for the JMS persistent stores:

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

   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. Create a new persistence store for the new BAMJMSServer, for example, **BAMJMSFileStore_N**. Specify the path for the store. The directory should be on shared storage as recommended in Section 4.3, "About Recommended Locations for the Different Directories." For example:

```
ORACLE_BASE/admin/domain_name/cluster_name/jms
```

> **Note:** You can also assign SOAJMSFileStore_N as store for the new BAM JMS Servers. For the purpose of clarity and isolation, individual persistent stores are used in the following steps.

f. Create a new JMS Server for BAM, for example, BAMJMSServer_N. Use the BAMJMSFileStore_N for this JMSServer. Target the BAMJMSServer_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 BAM JMS Modules 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 BAM: BAMJMSMOdule 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 BAMJMSServerXXXXXX, 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**.

4. Configuring Oracle Coherence for deploying composites for the new server as described in Section 9.4, "Configuring Oracle Coherence for Deploying Composites."

> **Note:** Only the **localhost** field must be changed for the server.
> Replace the localhost with the listen address of the new server added:
>
> ```
> Dtangosol.coherence.localhost=SOAHOST1VHNn
> ```

5. Reconfigure the JMS Adapter with the new server using the **FactoryProperties** field in the Administration Console. Click on the corresponding cell under the **Property** value and enter the following:

   ```
   java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory;java.naming.p
   rovider.url=t3://soahostvhn1:8001,soahos2tvhn1:8001;java.naming.security.princi
   pal=weblogic;java.naming.security.credentials=weblogic1
   ```

   Click **Save and Activate**.

6. Configure the persistent store for the new server. This should be a location visible from other nodes as recommended in Chapter 4, "Preparing the File System for an Enterprise Deployment."

   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. To configure server migration using the Oracle WebLogic Server Administration Console:

> **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.

  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 and click **Save**.

    This enables the Node Manager to start a failed server on the target node automatically.

  f. Restart the Administration Server, managed servers, and Node Manager.

    To restart the Administration Server, use the procedure in Section 8.4.3, "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 & 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 using the following command on the managed server PID:

```
kill -9 pid
```

You can identify the PID of the node using the following command:

```
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. Node Manager logs a message indicating that the server will not be restarted again locally.

16.5.3 Scale-up Procedure for Oracle BAM

You cannot scale a BAM Server managed server because the BAM Server runs in active-passive mode. However, you can scale a BAM Web Applications server.

There are two ways to scale a BAM Web Applications server:

- Clone the managed server that is running only the BAM Web Applications server.

- Clone the managed server that is running both the BAM Server and the BAM Web Applications Server and untarget the BAM server system as described in Section 12.6, "Untargeting the BAM Server System from WLS_BAM2."

To scale up a BAM Web Applications server, follow the steps in Section 16.5.1, "Scale-up Procedure for Oracle SOA" excluding Step 5, Configuring Oracle Coherence for deploying composites for the new server.

16.5.4 Scale-up Procedure for Oracle Service Bus

You can scale up the Oracle Service Bus servers by adding new managed servers to nodes that are already running one or more managed servers.

Prerequisites

Before scaling up you Oracle Service Bus servers, review the following prerequisites:

- You already have a cluster that runs managed servers configured with Oracle Service Bus components.

- The nodes contain Middleware home, an Oracle HOME (SOA and Oracle Service Bus) and a domain directory for existing managed servers.

- The source managed server you clone already exists on the node where you want to run the new managed server.

You can use the existing installations (the Middleware home, and domain directories) for creating new WLS_OSB servers. You do not need to install SOA or Oracle Service Bus binaries in a new location, or run pack and unpack.

To scale up the Oracle Service Bus servers:

1. Using the Administration Console, clone WLS_OSBn into a new managed server:

   a. Select **Environment** and then **Servers**.

   b. Select the managed server that you want to clone (for example, WLS_OSB1).

   c. Select **Clone**.

Name the new managed server **WLS_OSBn**, where n is a number to identify the new managed server.

For these steps you are adding a new server to SOAHOST1, which is already running WLS_OSB1.

2. For the listen address, assign the virtual host name to use for this new managed server.

   If you are planning to use server migration as recommended for this server, this virtual host name allows it to move to another node. The virtual host name should be different from those in use by other managed servers (it may be in the same or different domain) that are running in the nodes used by the Oracle Service Bus/SOA domain.

   To set the managed server listen address:

   a. Log into the Oracle WebLogic Server Administration Console.

   b. In the Change Center, click **Lock & Edit**.

   c. In the Domain Structure window expand the **Environment** node.

   d. Click **Servers**.

   The Summary of Servers page appears.

   e. In the **Names** column of the table, select the managed server with the listen address you want to update.

   The Settings page for that managed server appears.

   f. Set the Listen Address to **SOAHOST1VHNn** and click **Save**.

   Restart the managed server for the change to take effect.

3. Update the cluster address to include the new server:

   a. In the Administration console, select **Environment**, and then **Cluster**.

   b. Click the **OSB_Cluster** server.

   The Settings Screen for the OSB_Cluster appears.

   c. In the **Change Center**, click **Lock & Edit**.

   d. Add the new server's address and port to the **Cluster Address** field. For example:

   ```
   SOAHOST1VHN2:8011,SOAHOST2VHN2:8011,SOAHOST1VHNn:8011
   ```

4. If your Oracle Service Bus configuration includes one or more business services that use JMS request/response functionality, perform the following procedure using the Oracle Service Bus Console after adding the new managed server to the cluster:

   a. In the **Change Center**, click **Create** to create a session.

   b. Using the Project Explorer, locate and select a business service that uses JMS request/response.

   Business services of this type display Messaging Service as their Service Type.

   c. At the bottom of the View Details page, click **Edit**.

   d. If there is a cluster address in the endpoint URI, add the new server to the cluster address.

e. In the Edit a Business Service - Summary page, click **Save**.

f. Repeat the previous steps for each remaining business service that uses JMS request/response.

g. In the Change Center, click **Activate**.

h. Restart the managed server.

i. Restart the Administration Server.

The business services are now configured for operation in the extended domain.

Note: For business services that use a JMS MesageID correlation scheme, edit the connection factory settings to add an entry to the table mapping managed servers to queues. For information about configuring queues and topic destinations, see "JMS Server Targeting" in *Oracle Fusion Middleware Configuring and Managing JMS for Oracle WebLogic Server*.

---

5. If your Oracle Service Bus configuration includes one or more proxy services that use JMS endpoints with cluster addresses, perform the following procedure using the Oracle Service Bus Console after adding the new managed server to the cluster:

a. In the **Change Center**, click **Create** to create a session.

b. Using the Project Explorer, locate and select a proxy service that uses JMS endpoints with cluster addresses.

c. At the bottom of the View Details page, click **Edit**.

d. If there is a cluster address in the endpoint URI, add the new server to the cluster address.

e. On the Edit a Proxy Service - Summary page, click **Save**.

f. Repeat the previous steps for each remaining proxy service that uses JMS endpoints with cluster addresses.

g. In the **Change Center**, click **Activate**.

h. Restart the managed server.

The proxy services are now configured for operation in the extended domain.

6. Update the Oracle Service Bus result cache Coherence configuration for the new server:

a. Log into Oracle WebLogic Server Administration Console.

b. In the **Change Center**, click **Lock & Edit**.

c. In the **Domain Structure** window, expand the **Environment** node.

d. Click **Servers**.

The Summary of Servers page appears.

e. Click the name of the server (a hyperlink) in the **Name** column of the table.

The settings page for the selected server appears.

f. Click the **Server Start** tab.

Enter the following for WLS_OSBn (on a single line, without a carriage returns):

```
-DOSB.coherence.localhost=SOAHOST1vhnn -DOSB.coherence.localport=7890
-DOSB.coherence.wka1=SOAHOST1vhn2 -DOSB.coherence.wka1.port=7890
-DOSB.coherence.wka2=SOAHOST2vhn2 -DOSB.coherence.wka1.port=7890
```

Note: For this configuration servers WLS_OSB1 and WLS_OSB2 must be running (listening on Virtual Host Names SOAHOST1VHN and SOAHOST2VHN as used in the rest of the guide) when WLS_OSBn is started. This allows WLS_OSBn to join the coherence cluster started by either WLS_OSB1 or WLS_OSB2 using the WKA addresses specified. In addition, make sure WLS_OSB1 and WLS_OSB2 are started before WLS_OSBn is started when all three servers are restarted. This ensures WLS_OSBn joins the cluster started by one of WLS_OSB1 or WLS_OSB2. If the order in which the servers start is not important, add the host and port for WLS_OSBn as WKA for WLS_OSB1 and WLS_OSB2, and also add WLS_OSBn as WKA for WLS_OSBn.

---

   g. Save and activate the changes.

      Restart the Oracle Service Bus servers for the changes to take effect.

7. Reconfigure the JMS Adapter with the new server using the **FactoryProperties** field in the Administration Console. Click on the corresponding cell under the **Property** value and enter the following:

```
java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory;java.naming.p
rovider.url=t3://soahostvhn1:8001,soahos2tvhn1:8001;java.naming.security.princi
pal=weblogic;java.naming.security.credentials=weblogic1
```

Click **Save and Activate**.

8. Create JMS Servers and persistent stores for Oracle Service Bus reporting/internal destinations on the new managed server.

   a. Use the Oracle WebLogic Server Administration Console to create a new persistent store for the new WseeJMSServer and name it, for example, **OSB_rep_JMSFileStore_N**. Specify the path for the store. This should be a directory on shared storage, as recommended in Chapter 4, "Preparing the File System for an Enterprise Deployment." For example:

      ORACLE_BASE/admin/*DOMAIN_NAME*/*cluster_name*/jms/

      Target the store the new cloned server (WLS_OSBn).

   b. Create a new JMS Server for Oracle Service Bus, for example, OSB_rep_JMSServer_N. Use the OSB_rep_JMSFileStore_N for this JMSServer. Target the OSB_rep_JMSServer_N Server to the recently created Managed Server (WLS_OSBn)

   c. Update the SubDeployment targets for the "jmsResources" Oracle Service Bus JMS Module to include the recently created OSB JMS Server:

      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 **jmsResources** (a hyperlink in the **Names** column of the table). The Settings page for jmsResources appears.

Click the **SubDeployments** tab. The subdeployment module for jmsresources appears.

> **Note:** This subdeployment module name for destinations is a random name in the form of wlsbJMSServerXXXXXX resulting from the Configuration Wizard JMS configuration for the first two servers (WLS_OSB1 and WLS_OSB2).

Click the **wlsbJMSServerXXXXXX** subdeployment and update the targets to include the new OSB_rep_JMSServer_n server.

9. Create JMS Servers, persistent stores and destinations for OSB JAX-RPC on the new managed server.

> **Note:** WebLogic Advanced Web Services for JAX-RPC Extension uses regular (non-distributed) destinations to ensure that a locally processed request on a service gets enqueued only to a local member.

a. Use the Oracle WebLogic Server Administration Console to create a new persistent store for the new WseeJMSServer and name it, for example, **Wsee_ rpc_JMSFileStore_N**. Specify the path for the store. This should be a directory on shared storage, as recommended in Chapter 4, "Preparing the File System for an Enterprise Deployment."

b. Create a new JMS Server for OSB JAX-RPC, for example, OSB_rpc_JMSServer_ N. Use the Wsee_rpc_JMSFileStore_N for this JMSServer. Target the OSB_rpc_ JMSServer_N Server to the recently created Managed Server (WLS_OSBn).

c. Update the WseeJMSModule OSB JMS Module with destinations and the recently created OSB JMS Server by expanding 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 **WseeJmsModule** (a hyperlink in the **Names** column of the table). The Settings page for WseeJmsModule appears. Follow steps d through v to complete this step.

d. In the **Change Center**, click **Lock & Edit** and click **New**.

e. Select **DefaultCallbackQueue-WseeJmsServer_auto_n** and click **Next**.

f. Click **Create a New Subdeployment**.

g. Accept the default name and click **OK**.

h. Select **OSB_rpc_JMSServer_n** as the target and click **Finish**.

i. Update the local JNDI name for the destination:

In the **Change Center**, click **Lock & Edit**.

In the **Settings** for the WseeJmsModule page, click the **DefaultCallbackQueue-WseeJmsServer_auto_n** destination.

In the general **Configuration** tab, click **Advanced**.

Update the local JNDI name to **weblogic.wsee.DefaultCallbackQueue**.

    j. Repeat steps d through h for the **DefaultQueue-WseeJmsServer_auto_n queue**, using **weblogic.wsee.DefaultQueue-WseeJmsServer_auto_n** as the JNDi name and **weblogic.wsee.DefaultQueue** as the local JNDI name.

10. Create a new SAF agent and target it to the newly added managed server:

    a. In the Oracle WebLogic Server Administration Console, expand **Services**, **Messaging**, and then **Store-and-Forward Agents**

    b. Add a new SAF agent **ReliableWseeSAFAgent_auto_N**.

    c. Select persistent store **Wsee_rpc_JMSFileStore_N** (persistent store created for OSB JAX-RPC).

    d. Target the SAF Agent to the new managed server and activate changes.

11. Configure a TX persistent store for the new server in a location visible from the other nodes.

    a. From the Administration Console, select **Server_name** and then the **Services** tab.

    b. Under **Default Store**, in **Directory**, enter the path to the folder where you want the default persistent store to store its data files.

12. Disable host name verification for the new managed server. Before starting and verifying the WLS_OSBn managed server, 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 SOAHOSTn. You can ignore these steps if you have already disabled hostname verification for the source server from which the new server has been cloned (the hostname verification settings is propagated to the cloned server).

To disable host name verification:

    a. In the Oracle Enterprise Manager Console, select Oracle WebLogic Server Administration Console.

    b. Expand the **Environment** node in the **Domain Structure** window and click **Servers**.

    The Summary of Servers page appears.

    c. Select **WLS_OSBn** in the **Names** column of the table.

    The Settings page for the server appears.

    d. Click the **SSL** tab and click **Advanced**.

    e. Set **Hostname Verification** to **None** and click **Save**.

13. If it is not already started, start the Node Manager on the node. To start the Node Manager, use the installation in shared storage from the existing nodes as follows:

```
SOAHOSTN> WL_HOME/server/bin/startNodeManager
```

14. Start and test the new managed server from the Administration Console.

    a. Shut down the existing managed servers in the cluster.

    b. Ensure that the newly created managed server, **WLS_OSBn**, is up.

    c. Access the application on the newly created managed server using the following URL:

```
http://vip:port/sbinspection.wsil
```

15. Configure Server Migration for the new managed server.

> **Note:** Since this is a scale-up operation, the node should already contain a Node Manager and environment configured for server migration. The floating IP for the new Oracle Service Bus managed server should already be present.

To configure server migration:

a. Log into the Administration Console.

b. In the left pane, expand **Environment** and select **Servers**.

c. Select the name of the new managed server for which you want to configure migration.

d. Click the **Migration** tab.

e. In the **Available** field, in the **Migration Configuration** section, select the machines to which migration is allowed and click the right arrow.

f. Select the same migration targets used for the servers that already exist on the node.

For example, for new managed servers on SOAHOST1, which is already running WLS_OSB1, select SOAHOST2. For new managed servers on SOAHOST2, which is already running WLS_OSB2, select SOAHOST1.

Make sure the appropriate resources are available to run the managed servers concurrently during migration.

g. Select the **Automatic Server Migration Enabled** option and click **Save**.

This enables the Node Manager to start a failed server on the target node automatically.

h. Restart the Administration Server, managed servers, and Node Manager.

16. Test server migration for this new server from the node where you added the new server:

a. Stop the **WLS_OSBn** managed server by running the following command on the PID (process ID) of the managed server:

```
kill -9 pid
```

You can identify the PID of the node using the following command:

```
ps -ef | grep WLS_OSBn
```

Note: For Windows, you can terminate the Managed Server using the `taskkill` command. For example:

```
taskkill /f /pid pid
```

Where *pid* is the process ID of the Managed Server.

To determine the process ID of the WLS_OSBn Managed Server, run the following command:

```
MW_HOME\jrockit_160_20_D1.0.1-2124\bin\jps -l -v
```

---

b. In the Node Manager Console you can see a message appears indicating that WLS_OSBn's floating IP has been disabled.

c. Wait for the Node Manager to try a second restart of WLS_OSBn.

Node Manager waits for a fence period of 30 seconds before trying this restart.

d. Once Node Manager restarts the server, stop it again.

Node Manager logs a message indicating that the server will not be restarted again locally.

Note: After a server is migrated, to fail it back to its original node, 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.

16.6 Scaling Out the Topology (Adding Managed Servers to New Nodes)

When you scale 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, 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 the attachHome.sh script in the following location:

  ```
  ORACLE_HOME/oui/bin/
  ```

  To update the Middleware home list to add or remove a WL_HOME, edit the `beahomelist` file located in the following directory:

MW_HOME/bea

> **Note:** Scale out procedure apply only to WSM managed servers.

16.6.1 Scale-out Procedure for the Oracle SOA

To scale out the topology:

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 `beahomelist` file and add `MW_HOME` to it. The beahomelist file is located in the following directory:

   MW_HOME/bea

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_ WSMn, 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 virtual IP (also called a floating IP) for the server. This virtual IP 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 8.6, "Configuring the Java Object Cache for Oracle WSM."

9. Create JMS Servers for SOA and UMS on the new managed server.

   > **Note:** You do not have to create JMS servers for SOA and UMS on the new managed server if you are scaling up the WSM_WSM managed server or the BAM Web Applications system. This procedure is required only if you are scaling up the WLS_SOA managed servers

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 Chapter 4, "Preparing the File System for an Enterprise Deployment" as the directory for the JMS persistent stores:

    ORACLE_BASE/admin/domain_name/cluster_name/jms/

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 Chapter 4, "Preparing the File System for an Enterprise Deployment" as the directory for the JMS persistent stores:

    ORACLE_BASE/admin/domain_name/cluster_name/jms

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. Update the SubDeployment Targets for SOA, UMS and BPM JMS Modules (if applicable) to include the recently created JMS servers.

f. 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 the 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**.

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 and UMS 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 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, or UMSJMSServerXXXXXX, 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).

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

```
cd ORACLE_COMMON_HOME/common/bin

./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

```
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
```

Note: The configuration steps provided in this enterprise deployment topology are documented with the assumption that a local (per node) domain directory is used for each managed server.

---

11. Configuring Oracle Coherence for deploying composites for the new server as described in Section 9.4, "Configuring Oracle Coherence for Deploying Composites."

> **Note:** Only the **localhost** field needs to be changed for the server. Replace the localhost with the listen address of the new server added:
>
> ```
> Dtangosol.coherence.localhost=SOAHOST1VHNn
> ```

12. Reconfigure the JMS Adapter with the new server using the **FactoryProperties** field in the Administration Console. Click on the corresponding cell under the **Property** value and enter the following:

    ```
    java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory;java.naming.p
    rovider.url=t3://soahostvhn1:8001,soahos2tvhn1:8001;java.naming.security.princi
    pal=weblogic;java.naming.security.credentials=weblogic1
    ```

    Click **Save and Activate**.

13. Configure the persistent store for the new server. This should be a location visible from other nodes as recommended in Chapter 4, "Preparing the File System for an Enterprise Deployment."

    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.

14. 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**.

15. 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
    ```

16. 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 using the following URL:

  ```
  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 `soa_vh.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.

17. 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.

  To configure server migration:

  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 8.4.3, "Starting the Administration Server on SOAHOST1.".

18. 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 & 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.

19. Test server migration for this new server from the node where you added the new server:

    a. Abruptly stop the WLS_SOA*n* managed server by running the following command;

    ```
    kill -9 pid
    ```

    You can identify the PID (process ID) of the node using the following command:

    ```
    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.

16.6.2 Scaling out the BAM Topology

To scale out the topology:

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 `beahomelist` file and add `MW_HOME` to it. The `beahomelist` file is located in the following directory:

    ```
    MW_HOME/bea
    ```

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 into a new managed server. Name it WLS_SOAn, 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 virtual IP (also called a floating IP) for the server. This virtual IP should be different from the one used for the existing managed server.

8. Create JMS Servers for SOA, BAM, (if applicable) and UMS on the new managed server.

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 Chapter 4, "Preparing the File System for an Enterprise Deployment" as the directory for the JMS persistent stores:

```
ORACLE_BASE/admin/domain_name/cluster_name/jms/
```

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 Chapter 4, "Preparing the File System for an Enterprise Deployment" as the directory for the JMS persistent stores:

```
ORACLE_BASE/admin/domain_name/cluster_name/jms
```

> **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. Create a new persistence store for the new BAMJMSServer, for example, **BAMJMSFileStore_N**. Specify the path for the store. This should be a directory on shared storage as recommended in Chapter 4, "Preparing the File System for an Enterprise Deployment."

```
ORACLE_BASE/admin/domain_name/cluster_name/jms
```

> **Note:** You can also assign SOAJMSFileStore_N as store for the new BAM JMS Servers. For the purpose of clarity and isolation, individual persistent stores are used in the following steps.

f. Create a new JMS Server for BAM, for example, BAMJMSServer_N. Use the BAMJMSFileStore_N for this JMSServer. Target the BAMJMSServer_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 SOAJMSModule (represented as a hyperlink in the Names column of the table). The Settings page for SOAJMSModule 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 BAM 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 BAM: BAMJMSMOdule 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 BAMJMSServerXXXXXX, 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).

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

```
cd ORACLE_COMMON_HOME/common/bin

./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

```
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
```

> **Note:** The configuration steps provided in this enterprise deployment topology are documented with the assumption that a local (per node) domain directory is used for each managed server.

10. Configuring Oracle Coherence for deploying composites for the new server as described in Section 9.4, "Configuring Oracle Coherence for Deploying Composites."

> **Note:** Only the **localhost** field needs to be changed for the server. Replace the localhost with the listen address of the new server added:
>
> ```
> Dtangosol.coherence.localhost=SOAHOST1VHNn
> ```

11. Reconfigure the JMS Adapter with the new server using the **FactoryProperties** field in the Administration Console. Click on the corresponding cell under the **Property** value and enter the following:

```
java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory;java.naming.p
rovider.url=t3://soahostvhn1:8001,soahos2tvhn1:8001;java.naming.security.princi
pal=weblogic;java.naming.security.credentials=weblogic1
```

Click **Save and Activate**.

12. Configure the persistent store for the new server. This should be a location visible from other nodes as recommended in Chapter 4, "Preparing the File System for an Enterprise Deployment."

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 using the following URL:

```
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 `soa_vh.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.

To configure server migration:

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 8.4.3, "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 & 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 from the node where you added the new server:

a. Abruptly stop the WLS_SOA*n* managed server by running the following command:

```
kill -9 pid
```

You can identify the PID (process ID) of the node using the following command:

```
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.

16.6.3 Scale-out Procedure for Oracle BAM

You cannot scale a BAM Server managed server because the BAM Server runs in active-passive mode. However, you can scale a BAM Web Applications server.

There are two ways to scale a BAM Web Applications server:

■ Clone the managed server that is running only the BAM Web Applications server.

■ Clone the managed server that is running both the BAM Server and the BAM Web Applications Server and untarget the BAM server system as described in Section 12.6, "Untargeting the BAM Server System from WLS_BAM2."

To scale out a BAM Web Applications server, follow the steps in Section 16.6.1, "Scale-out Procedure for the Oracle SOA" excluding Step 11, Configuring Oracle Coherence for deploying composites for the new server.

16.6.4 Scale-out Procedure for Oracle Service Bus

When you scale out the topology, you add new managed servers configured with Oracle Service Bus to the new nodes.

Prerequisites

Before scaling out the Oracle Service Bus topology, make sure you meet these prerequisites:

■ There must be existing nodes running managed servers configured with Oracle Service Bus within the topology.

■ The new node optionally can access the existing home directories for WebLogic Server and Oracle Service Bus installation. Use the existing installations in shared storage for creating a new WLS_OSB managed server. You do not need to install WebLogic Server or Oracle Service Bus binaries in every new location in this case, but you do need to run the pack and unpack commands to bootstrap the domain configuration in the new node, unless you are scaling the Oracle Service Bus server to machines containing other servers of the same domain (the SOA servers).

■ If there is no existing installation in shared storage, install WebLogic Server, SOA, and Oracle Service Bus in the new nodes.

■ When multiple servers in different nodes share an ORACLE_HOME or WL_HOME, 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 the attachHome.sh file located in the following directory:

ORACLE_HOME/oui/bin/

To update the Middleware home list to add or remove a WL_HOME, edit the `beahomelist` file located in the following directory:

MW_HOME/bea

To scale out the topology:

1. On the new node, mount the existing Middleware Home. It should include the Oracle Service Bus and SOA (if homes are shared) installation and the domain directory. Ensure that the new node has access to this directory, just like other nodes in the domain.

2. Attach ORACLE_HOME in shared storage to the local Oracle Inventory using the following command:

   ```
   SOAHOSTn>cd ORACLE_BASE/product/fmw/soa/
   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 `beahomelist` file located in the following directory:

   ```
   MW_HOME/bea/
   ```

   Add `ORACLE_BASE/product/fmw` to the list.

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_OSB1 into a new managed server. Name it **WLS_OSBn**, where *n* is a number, and assign it to the new machine.

   > **Note:** For these steps, you are adding a new server to node *n*, where no managed server was running previously.

7. For the listen address, assign the virtual host name to use for this new managed server. If you are planning to use server migration as recommended for this server, this virtual host name allows it to move to another node. The virtual host name should be different from those used by other managed servers (may be in the same or different domain) that are running in the nodes used by the OSB/SOA domain.

   a. Log into the Oracle WebLogic Server Administration Console.

   b. In the **Change Center**, click **Lock & Edit**.

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

   d. Click **Servers**.

      The Summary of Servers page appears.

   e. Select the managed server with listen addresses you want to update in the **Names** column of the table.

      The Setting page for that managed server appears.

    f. Set the Listen Address to **SOAHOSTnVHN1** and click **Save**.

    g. Save and activate the changes.

    h. Restart the managed server.

8. Update the cluster address to include the new server:

    a. Select **Environment**, and then **Cluster** from the Administration Console.

    b. Click the **OSB_Cluster** server.

    The Settings Screen for the OSB_Cluster appears.

    c. In the **Change Center**, click **Lock & Edit**.

    d. Add the new server's address and port to the **Cluster Address** field. For example:

```
SOAHOST1VHN1:8011,SOAHOST2VHN1:8011,SOAHOSTNVHN1:8011
```

9. Create JMS servers and persistent stores for Oracle Service Bus reporting/internal destinations on the new managed server.

    a. Use the Oracle WebLogic Server Administration Console to create a new persistent store for the new **WseeJMSServer** and name it, for example, **OSB_rep_JMSFileStore_N**. Specify the path for the store. This should be a directory on shared storage as recommended in Chapter 4, "Preparing the File System for an Enterprise Deployment."

> **Note:** This directory must exist before the managed server is started or the start operation fails.
>
> *ORACLE_BASE*/admin/*domain_name*/*cluster_name*/jms/OSB_rep_JMSFileStore_N

---

    b. Create a new JMS Server for Oracle Service Bus, for example, **OSB_rep_JMSServer_N**. Use the **OSB_rep_JMSFileStore_N** for this JMSServer. Target the **OSB_rep_JMSServer_N** Server to the recently created managed server (**WLS_OSBn**).

    c. Update the **SubDeployment** targets for the **jmsresources** Oracle Service Bus JMS Module to include the recently created Oracle Service Bus JMS Server:

    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 **jmsresources** (a hyperlink in the **Names** column of the table). The Settings page for jmsResources appears.

    Open the **SubDeployments** tab. The subdeployment module for jmsresources appears.

> **Note:** This subdeployment module name is a random name in the form of `wlsbJMSServerXXXXXX` resulting from the Configuration Wizard JMS configuration for the first two servers (WLS_OSB1 and WLS_OSB2).
>
> Click the **wlsbJMSServerXXXXXX** subdeployment and update the targets to include the new **OSB_rep_JMSServer_N** server.

10. Create JMS Servers, persistent stores and destinations for OSB JAX-RPC on the new managed server.

> **Note:** WebLogic Advanced Web Services for JAX-RPC Extension uses regular (non-distributed) destinations to ensure that a locally processed request on a service gets enqueued only to a local member.

a. Use the Oracle WebLogic Server Administration Console to create a new persistent store for the new WseeJMSServer and name it, for example, **Wsee_rpc_JMSFileStore_N**. Specify the path for the store. This should be a directory on shared storage as recommended in Chapter 4, "Preparing the File System for an Enterprise Deployment."

> **Note:** This directory must exist before the managed server is started or the start operation fails.
>
> ```
> ORACLE_BASE/admin/DOMAIN_NAME/cluster_name/jms/Wsee_rpc_
> JMSFileStore_N
> ```

b. Create a new JMS Server for Oracle Service Bus JAX-RPC, for example, **OSB_rpc_JMSServer_N**. Use the **Wsee_rpc_JMSFileStore_N** for this JMSServer. Target the **OSB_rpc_JMSServer_N** Server to the recently created Managed Server (**WLS_OSBn**).

c. Update the WseeJMSModule Oracle Service Bus JMS Module with destinations and the recently created Oracle Service Bus JMS Server:

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 **WseeJmsModule** (a hyperlink in the **Names** column of the table). The Settings page for WseeJmsModule appears.

Follow steps d through j to complete this step.

d. In the **Change Center**, click **Lock & Edit** and click **New**.

e. Select **Queue** and click **Next**.

f. Enter **DefaultCallbackQueue-WseeJmsServer_auto_n** as name for the queue.

g. Enter **weblogic.wsee.DefaultCallbackQueue-WseeJmsServer_auto_n** as the JNDI name and click **Next**.

h. Click **Create a New Subdeployment**.

i. Accept the default name and click **OK**.

j. Select **OSB_rpc_JMSServer_n** as the target and click **Finish**.

k. Activate the changes.

l. Update the local JNDI name for the destination:

In the **Change Center**, click **Lock & Edit**.

In the **Settings** page for WseeJmsModule, click the **DefaultCallbackQueue-WseeJmsServer_auto_n** destination.

In the general **Configuration** tab, click **Advanced**.

Update the local JNDI name to **weblogic.wsee.DefaultCallbackQueue**.

11. Create a new SAF agent and target it to the newly added managed server:

In the Oracle WebLogic Server Administration Console, expand **Services**, **Messaging** and then **Store-and-Forward Agents**, and add a new SAF agent, **ReliableWseeSAFAgent_auto_N**.

Select persistent store **Wsee_rpc_JMSFileStore_N** (persistent store created for Oracle Service Bus JAX-RPC). Target the SAF Agent to the new managed server and activate changes.

12. If your Oracle Service Bus configuration includes one or more business services that use JMS request/response functionality, follow this procedure using the Oracle Service Bus Console after adding the new managed server to the cluster:

a. In the **Change Center**, click **Create** to create a session.

b. Using the Project Explorer, locate and select a business service that uses JMS request/response.

Business services of this type display Messaging Service as their Service Type.

c. At the bottom of the **View Details** page, click **Edit**.

d. If there is a cluster address in the endpoint URI, add the new server to the cluster address.

e. On the Edit a Business Service - Summary page, click **Save**.

f. Repeat the previous steps for each remaining business service that uses JMS request/response.

g. In the **Change Center**, click **Activate**.

h. Restart the managed server.

i. Restart the Administration Server.

The business services are now configured for operation in the extended domain.

Note: For business services that use a JMS MesageID correlation scheme, edit the connection factory settings to add an entry to the table mapping managed servers to queues. For information on how to configure queues and topic destinations, see "JMS Server Targeting" in *Oracle Fusion Middleware Configuring and Managing JMS for Oracle WebLogic Server*.

---

13. If your Oracle Service Bus configuration includes one or more proxy services that use JMS endpoints with cluster addresses, perform the following procedure using

the Oracle Service Bus Console after adding the new managed server to the cluster:

a. In the **Change Center**, click **Create** to create a session.

b. Using the Project Explorer, locate and select a proxy service that uses JMS endpoints with cluster addresses.

c. At the bottom of the View Details page, click **Edit**.

d. If there is a cluster address in the endpoint URI, add the new server to the cluster address.

e. On the Edit a Proxy Service - Summary page, click **Save**.

f. Repeat the previous steps for each remaining proxy service that uses JMS endpoints with cluster addresses.

g. In the **Change Center**, click **Activate**.

h. Restart the managed server.

The proxy services are now configured for operation in the extended domain.

14. Update the Oracle Service Bus result cache Coherence configuration for the new server:

a. Log into Oracle WebLogic Server Administration Console. In the **Change Center**, click **Lock & Edit**.

b. In the **Domain Structure** window, expand the **Environment** node.

c. Click **Servers**.

The Summary of Servers page appears.

d. Click the name of the server (a hyperlink) in the **Name** column of the table.

The settings page for the selected server appears.

e. Click the **Server Start** tab.

f. Click **Advanced**.

g. Enter the following for WLS_OSBn (on a single line, without a carriage returns):

```
-DOSB.coherence.localhost=SOAHOSTnvhn1 -DOSB.coherence.localport=7890
-DOSB.coherence.wka1=SOAHOST1vhn1 -DOSB.coherence.wka1.port=7890
-DOSB.coherence.wka2=SOAHOST2vhn1 -DOSB.coherence.wka1.port=7890
```

Note: For the previous configuration, servers WLS_OSB1 and WLS_OSB2 are running when WLS_OSBn starts. This allows WLS_OSBn to join the coherence cluster started by either WLS_OSB1 or WLS_OSB2 using the WKA addresses specified. In addition, make sure WLS_OSB1 and WLS_OSB2 are started before WLS_OSBn is started when starting all three servers. This ensures WLS_OSBn joins the cluster started by either WLS_OSB1 or WLS_OSB2. For a configuration where the order in which the servers are started does not matter, add the host and port for WLS_OSBn as WKA for WLS_OSB1 and WLS_OSB2, and also add WLS_OSBn as WKA for WLS_OSBn.

---

h. Save and activate the changes

Restart the Oracle Service Bus servers.

15. Reconfigure the JMS Adapter with the new server using the **FactoryProperties** field in the Administration Console. Click on the corresponding cell under the **Property** value and enter the following:

```
java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory;java.naming.p
rovider.url=t3://soahostvhn1:8001,soahos2tvhn1:8001;java.naming.security.princi
pal=weblogic;java.naming.security.credentials=weblogic1
```

Click **Save and Activate**.

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

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

```
./pack.sh -managed=true -domain=MW_HOME/user_projects/domains/soadomain/
-template=soadomaintemplateScale.jar -template_name=soa_domain_templateScale
```

Run the following command on SOAHOST1 to copy the template file created to SOAHOSTn:

```
scp soadomaintemplateScale.jar oracle@SOAHOSTN:/ORACLE_
BASE/product/fmw/soa/common/bin
```

Run the unpack command on SOAHOST*N* to unpack the template in the managed server domain directory as follows:

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

```
./unpack.sh -domain=ORACLE_BASE/product/fmw/user_projects/domains/soadomain/
-template=soadomaintemplateScale.jar
```

> **Note:** The configuration steps provided in this enterprise deployment topology are documented with the assumption that a local (per node) domain directory is used for each managed server.

---

17. Configure a TX persistent store for the new server in a location visible from other nodes as indicated in the recommendations about shared storage

   a. From the Administration Console, select the server name, and then the **Services** tab.

   b. Under **Default Store**, in **Directory**, enter the path to the folder where you want the default persistent store to store its data files.

18. Disable host name verification for the new managed server.

Before starting and verifying the WLS_OSBn managed server, 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 SOAHOSTn. If you have already disabled host name verification for the source server from which the new server has been cloned, you can skip this procedure (the hostname verification setting is propagated to the cloned server).

To disable host name verification:

   a. In the Oracle Enterprise Manager Console, select Oracle WebLogic Server Administration Console.

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

c. Click **Servers**.

The Summary of Servers page appears.

d. Select **WLS_OSBn** 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** and click **Save**.

19. Start the Node Manager on the new node using the installation in shared storage from the existing nodes. Pass the host name of the new node as a parameter:

```
SOAHOSTN> WL_HOME/server/bin/startNodeManager new_node_ip
```

20. Start and test the new managed server from the Oracle WebLogic Server Administration Console:

a. Shut down all the existing managed servers in the cluster.

b. Ensure that the newly created managed server, WLS_OSBn, is running. Access the application on the newly created managed server:

```
http://vip:port/sbinspection.wsil
```

The application should be functional.

21. Configure server migration for the new managed server.

> **Note:** Since this new node is using an existing shared storage installation, it is already 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 Oracle Service Bus Managed Server is already present in the new node.

To configure server migration:

a. Log into the Administration Console.

b. In the left pane, expand **Environment** and select **Servers**.

c. Select the server (represented as hyperlink) for which you want to configure migration from the **Names** column of the table.

The Settings page for that server appears.

d. Click the **Migration** tab.

e. In the **Available** field, in the **Migration Configuration** section, select the machines to which the server is to be migrated migration and click the right arrow.

For example, for new managed servers on SOAHOST1, which is already running WLS_OSB1, select **SOAHOST2**. For new managed servers on SOAHOST2, which is already running WLS_OSB2, select **SOAHOST1**.

> **Note:** Specify the least-loaded machine as the migration target for the new server. Complete the required capacity planning so that this node has enough available resources to sustain an additional managed server.

    f. Select the **Automatic Server Migration Enabled** option and click **Save**.

    This enables the Node Manager to start a failed server on the target node automatically.

    g. Restart the Administration Server, managed servers, and Node Manager.

22. Test server migration for this new server from the node where you added the new server:

    a. Abruptly stop the WLS_OSBn managed server by running the following command on the PID (process ID) of the managed server:

```
kill -9 pid
```

    You can identify the PID of the node using the following command:

```
ps -ef | grep WLS_OSBn
```

> **Note:** For Windows, you can terminate the managed server using the `taskkill` command. For example:
>
> ```
> taskkill /f /pid pid
> ```
>
> Where *pid* is the process Id of the managed server.
>
> You can determine the process ID of the WLS_OSBn managed server using the following command:
>
> ```
> MW_HOME\jrockit_160_20_D1.0.1-2124\bin\jps -l -v
> ```

    b. In the Node Manager Console you can view a message indicating that WLS_OSBn's floating IP has been disabled.

    c. Wait for the Node Manager to try a second restart of WLS_OSBn. 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 logs a message indicating that the server will not be restarted again locally.

> **Note:** After a server is migrated, to fail it back to its original node/machine, stop the managed server from the Oracle WebLogic Administration Console and then start it again. The appropriate Node Manager will start the managed server on the machine to which it was originally assigned.

16.7 Verifying Manual Failover of the Administration Server

In case a node fails, you can fail over the Administration Server to another node. The following sections provide the steps to verify the failover and failback of the Administration Server from SOAHOST1 and SOAHOST2.

Assumptions:

- The Administration Server is configured to listen on ADMINVHN, and not on ANY address. See step 14 in Section 8.3, "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

  - ADMINVHN : 100.200.140.206. This is the Virtual IP 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.

- Oracle WebLogic Server and Oracle Fusion Middleware components have been installed in SOAHOST2 as described in Chapter 6, "Installing the Software for an Enterprise Deployment" (that is, the same paths for ORACLE_HOME and MW_ HOME that exist on SOAHOST1 are also available on SOAHOST2)

This section contains the following topics:

- Section 16.7.1, "Failing Over the Administration Server to a Different Node"

- Section 16.7.2, "Validating Access to SOAHOST2 Through Oracle HTTP Server"

- Section 16.7.3, "Failing the Administration Server Back to SOAHOST1"

16.7.1 Failing Over the Administration Server to a Different Node

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).

To fail over the Administration Server to a different node:

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):

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

   b. Run the following command on SOAHOST2:

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

      For example:

```
/sbin/ifconfig eth0:1 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:

```
/sbin/arping -q -U -c 3 -I eth0 10.0.0.1
```

4. Start the Administration Server on SOAHOST2 using the procedure in Section 8.4.3, "Starting the Administration Server on SOAHOST1.".

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

    a. Ensure that you can access the Oracle WebLogic Server Administration Console using the following URL:

    ```
    http://ADMINVHN:7001/console
    ```

    b. Check that you can access and verify the status of components in the Oracle Enterprise Manager using the following URL:

    ```
    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.

16.7.2 Validating Access to SOAHOST2 Through Oracle HTTP Server

Perform the same steps as in Section 8.7.5, "Validating Access Through Oracle HTTP Server". This is to check that you can access the Administration Server when it is running on SOAHOST2.

16.7.3 Failing the Administration Server Back to SOAHOST1

This step checks that you can fail back the Administration Server, that is, stop it on SOAHOST2 and run it on SOAHOST1 by migrating ADMINVHN back to SOAHOST1 node.

To migrate ADMINVHN back to SOAHOST1:

1. Make sure the Administration Server is not running.

2. Run the following command on SOAHOST2.

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

3. Run the following command on SOAHOST1:

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

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

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

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

5. Start the Administration Server again on SOAHOST1 using the procedure in Section 8.4.3, "Starting the Administration Server on SOAHOST1."

   ```
   cd ORACLE_BASE/admin/domain_name/aserver/domain_name/bin
   ./startWebLogic.sh
   ```

6. Test that you can access the Oracle WebLogic Server Administration Console using the following URL:

   ```
   http://ADMINVHN:7001/console
   ```

7. Check that you can access and verify the status of components in the Oracle Enterprise Manager using the following URL:

   ```
   http://ADMINVHN:7001/em
   ```

16.8 Performing Backups and Recoveries in the SOA Enterprise Deployments

For information about backing up the environment, see "Backing Up Your Environment" in the *Oracle Fusion Middleware Administrator's Guide*. For information about recovering your information, see "Recovering Your Environment" in the *Oracle Fusion Middleware Administrator's Guide*.

Table 16–1 lists the static artifacts to back up in the 11*g* SOA enterprise deployment.

Table 16–1   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, `MW_HOME/bea/beahomelist`, `oraInst.loc`, `oratab`	N/A

Table 16–2 lists the runtime artifacts to back up in the 11*g* SOA enterprise deployment.

Table 16–2   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

> **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 in the following directory
>
> *ORACLE_HOME*/soa/thirdparty/edifecs/XEngine
>
> To back up ORACLE_HOME:
>
> tar -cvpf fmwhomeback.tar *MW_HOME*

16.9 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 virtual IPs 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 sqlnet.ora file, located in the following directory:

ORACLE_HOME/network/admin

The n indicates the time in minutes. Set this value to less than the known value of the timeout for the network device (that is, a firewall). For Oracle RAC, set this parameter in all of the Oracle home directories.

16.10 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.

16.11 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 16–1.

Example 16–1   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.

16.12 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 the *Oracle Fusion Middleware Administrator's
Guide for Oracle SOA Suite*.

16.13 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 the following location as
for the deployment plan and applications:

ORACLE_BASE/admin/*domain_name*/*cluster_name*/dp

This directory must be accessible from all nodes in the Enterprise Deployment
topology, as recommended in Chapter 4, "Preparing the File System for an Enterprise
Deployment."

16.14 Troubleshooting the Topology in an Enterprise Deployment

This section describes possible issues with the SOA enterprise deployment and
suggested solutions.

This section covers the following topics:

16.14.1 Access to BAM Results in HTTP Error 404

Problem: Accessing the BAM application results in the HTTP 404 error ("Not Found"). Starting the BAM server before the start of the database instance where BAM schemas reside may be the cause of this error.

Solution: Shut down the BAM instance and restart it after ensuring that the database is already up.

16.14.2 Page Not Found When Accessing soa-infra Application Through Load Balancer

Problem: You receive a `404` `"page not found"` message 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.

16.14.3 Error While Retrieving Oracle B2B Document Definitions

Problem: You receive an error when trying to retrieve a document definition XSD from Oracle B2B. B2B resides in a cluster and is accessed through a load balancer. The B2B console reports the following:

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

Solution: Set the frontend HTTP host and port for the Oracle WebLogic cluster where Oracle B2B resides. 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 and click **Save**:

   ■ **Frontend Host:** `soa.mycompany.com`

   ■ **Frontend HTTPS Port:** `443`

   ■ **Frontend HTTP Port:** `80`

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

7. Restart the servers to make the Frontend Host directive in the cluster effective.

16.14.4 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.

16.14.5 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. 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: Edit the `<jazn-policy>` element the `system-jazn-data.xml` file to grant permission to `BAM-services.jar`:

```
<grant>
  <grantee>
    <codesource>
<url>file:${oracle.home}/soa/modules/oracle.soa.workflow_11.1.1/BAM-
services.jar</url>
    </codesource>
  </grantee>
  <permissions>
    <permission>
      <class>java.security.AllPermission</class>
```

```
        </permission>
      </permissions>
    </grant>
```

16.14.6 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.

16.14.7 Administration Server Fails to Start After a Manual Failover

Problem: the Administration Server fails to start after it fails and you performed a manual failover to another node. 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: Remove the file `EmbeddedLDAP.lok` file from the following directory:

```
ORACLE_BASE/admin/domain_name/aserver/domain_
name/servers/AdminServer/data/ldap/ldapfiles/
.
```

16.14.8 Error While Activating Changes in Administration Console

Problem: Activation of changes in Administration Console fails after you have made changes to a server's start configuration. 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: 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).

16.14.9 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 virtual IP 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 virtual IP in any interface). Executing the command "sudo ifconfig $INTERFACE $ADDRESS $NETMASK" does not enable the IP in the failover node.

Solution: The rights and configuration for `sudo` execution should not prompt for a password. Verify the configuration of `sudo` with your system administrator so that `sudo` works without a password prompt.

16.14.10 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 virtual IP has been migrated and the server started. The virtual IP used by the SOA Server cannot be pinged from the client's node (that is, the node where the browser is being used).

Solution: Update the `nodemanager.properties` file to include the MACBroadcast or execute a manual arping:

```
/sbin/arping -b -q -c 3 -A -I $INTERFACE $ADDRESS > $NullDevice 2>&1
```

Where `$INTERFACE` is the network interface where the Virtual IP is enabled and `$ADDRESS` is the virtual IP address.

16.14.11 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.

16.14.12 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 typically succeeds. There is no cleanup or any other additional steps required.

16.14.13 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.

16.14.14 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. You do not have to log in again. Update the following URL and directly access the home page for the Administration Console:

```
soa.mycompany.com/console/console.portal
```

> **Note:**   This problem does not occur if you disable tracking of the changes described in this section.

16.14.15 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.

16.14.16 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.

16.14.17 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 following directory and restart the server:

```
OARCLE_BASE/admin/domain_name/mserver/domain_name/servers/server_
name/data/nodemanager
```

16.14.18 Configuring JOC for B2B Delivery Channel Updates

The default MDS change notification mechanisms in a SOA cluster propagate updates with a default frequency of 30 seconds. For faster propagation of configuration changes, such as delivery channels for a B2B agreement, and for cases where these changes are frequent, you can use Java Object Cache (JOC). Configure the distributed Java Object Cache using the `configure-joc.py` script located in the following directory:

MW_HOME/oracle_common/bin

Use this Python script to configure JOC in the managed servers for quick notification of changes made in B2B delivery channels. The script runs in WLST online mode. The Administration Server must be up and running.

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

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

To configure the distributed Java Object Cache for Oracle SOA Suite Servers:

1. Connect to the Administration Server using the command-line Oracle WebLogic Scripting Tool, `wlst.sh`, in the following directory:

    MW_HOME/oracle_common/common/bin

    To connect using `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 in a given cluster.

    Enter **y** when the script prompts you for whether you want to specify a cluster name. Also, specify the cluster name and discover port, when prompted. This discovers all the managed servers for the given cluster and configures the JOC. The discover port is common for the entire JOC configuration across the cluster.

    ```
    For Oracle Web Services Manager: Do you want to specify a cluster name (y/n)
    <y>
    Enter Cluster Name : SOA_Cluster
    Enter Discover Port : 9992
    ```

The following is an example of the configure-joc.py for high availability environments:

```
execfile('MW_HOME/oracle_common/bin/configure-joc.py')
.
Enter Hostnames (eg host1,host2) : SOAHOST1VHN1,SOAHOST2VHN1
.
Do you want to specify a cluster name (y/n) <y>y
.
Enter Cluster Name : SOA_Cluster
.
Enter Discover Port : 9992
.
Enter Distribute Mode (true|false) <true> : true
.
Do you want to exclude any server(s) from JOC configuration (y/n) <n> n
```

You can also use the script for 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 (WLS_WSM1:9998, WLS_WSM1:9998) : WLS_
  WSM1:9992,WLS_WSM2:9992
  ```

- 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'.

- Modify the `javacahce.xml` file to use the B2B server's VHN as the listen address:

  Edit the `javacache.xml` file for the server in question. This file is located in the following directory:

  DOMAIN_HOME/aserver/*soaedg_domain*/config/fmwconfig/servers/*server_name*

  Add the listen-address field as follows:

  ```
  ...
  <packet-distributor enable-router="false" startable="true"
  dedicated-coordinator="false">
              <listener-address host="SOAHOST1VHN1" port="9992" />
  <listener-address host="SOAHOST1VHN1" port="9992" />
              <distributor-location host=" SOAHOST1VHN1" port="9992" ssl="true"/>
  </packet-distributor>
  ...
  ```

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*.

16.14.19 SOA Coherence Cluster Conflicts when Multiple Clusters Reside in the Same Node

Problem: soa-infra fails to come up when multiple soa clusters reside in the same nodes. Messages such as the following appear in the server's .out file:

```
<Error> <Coherence> <BEA-000000> <Oracle Coherence GE 3.6.0.4 <Error>
(thread=Cluster, member=1): This senior Member(…) appears to have been
disconnected from another senior Member…stopping cluster service.>
```

Solution: When a Coherence member restarts, it attempts to bind to the port configured in its localport setting. If this port is not available, it increments the port

number (by two) and attempts to connect to that port. If multiple SOA clusters use similar range ports for coherence it is possible for a member to join a cluster with a different WKA, causing conflicts and preventing soa-infra application from starting. There are several ways to resolve this issue:

- Set up a port range for each of the various clusters instead of incrementing the cluster port by 2. For example, 8000-8090 for cluster 1, 8091-8180 for cluster 2. This is implicit in the model recommended in this guide specified in Table 3–4 where different ranges should be used for each coherence cluster.

- Disable port auto adjust to force the members to use their configured localhost address. This can be done via system property "tangosol.coherence.localport.adjust" for example -Dtangosol.coherence.localport.adjust=false.

- Configure a unique cluster name for each cluster. This can be done using the system property `tangosol.coherence.cluster`. For example:

  ```
  -Dtangosol.coherence.cluster=SOA_Cluster1
  ```

For more information on these different options, refer to the coherence cluster configuration documentation at the following URL:

```
http://download.oracle.com/docs/cd/E24290_
01/coh.371/e22837/cluster_setup.htm
```

16.14.20 Sudo Error Occurs During Server Migration

Problem: When running `wlsifconfig` for server migration, the following warning displays:

```
sudo: sorry, you must have a tty to run sudo
```

Solution: The WebLogic user ('oracle') is not allowed to run sudo in the background. To solve this, add the following line into `/etc/sudoers`:

```
Defaults:oracle !requiretty
```

See also, Section 14.5, "Setting Environment and Superuser Privileges for the wlsifconfig.sh Script".

16.14.21 Transaction Timeout Error

Problem: The following transaction timeout error appears in the log:

```
Internal Exception: java.sql.SQLException: Unexpected exception while enlisting
 XAConnection java.sql.SQLException: XA error: XAResource.XAER_NOTA start()
failed on resource 'SOADataSource_soaedg_domain': XAER_NOTA : The XID
is not valid
```

Solution: Check your transaction timeout settings, and be sure that the JTA transaction time out is less than the DataSource XA Transaction Timeout, which is less than the `distributed_lock_timeout` (at the database).

With the out of the box configuration, the SOA datasources do not set XA timeout to any value. The `Set XA Transaction Timeout` configuration parameter is unchecked in the WebLogic Server Administration Console. In this case, the datasources use the domain level JTA timeout which is set to `30`. Also, the default `distributed_lock_timeout` value for the database is `60`. As a result, the SOA configuration works correctly for any system where transactions are expected to have

lower life expectancy than such values. Adjust these values according to the transaction times your specific operations are expected to take.

16.14.22 Exceeded Maximum Size Error Messages

Problem: When complex rules are edited and saved in a SOA cluster, error messages reporting exceeded maximum size in replication messages may show up in the server's out file. For example:

```
<rws3211539-v2.company.com> <WLS_SOA1> <ExecuteThread: '2' for queue:
'weblogic.socket.Muxer'> <<WLS Kernel>> <> <> <1326464549135> <BEA-000403>
<IOException occurred on socket:
Socket[addr=/10.10.10.10,port=48290,localport=8001]
weblogic.socket.MaxMessageSizeExceededException: Incoming message of size:
'10000080' bytes exceeds the configured maximum of: '10000000' bytes for
protocol: 't3'.
weblogic.socket.MaxMessageSizeExceededException: Incoming message of size:
'10000080' bytes exceeds the configured maximum of: '10000000' bytes for
protocol: 't3'
```

Solution: This error is due to the large size of serialized rules placed in the HTTP Sessions and replicated in the cluster. Increase the maximum message size according the size of the rules being used.

To increase the maximum message size:

1. Log in to the WebLogic Administration Console.

2. Select **Servers**, **Server_name**, **Protocols**, and then **General**.

3. Modify the **Maximum Message Size** field as needed.

16.14.23 Out of Sync Clocks Cause Error

Problem: Servers that are configured for server migration and are using an Oracle RAC database for leasing shut themselves down, reporting messages such as the following:

```
<Error> <Cluster> <BEA-000150> <Server failed to get a connection to the
database in the past 60 seconds for lease renewal. Server will shut itself
down.>
<Critical> <Health> <BEA-310006> <Critical Subsystem ServerMigration has
failed. Setting server state to FAILED. Reason: ServerXXX failed to renew
lease in the database>
 <Critical> <WebLogicServer> <BEA-000385><Server health failed. Reason:
health of critical service 'ServerMigration'failed>
<Notice> <WebLogicServer> <BEA-000365> <Server state changed to FAILED>
```

Despite the message, the datasource used for leasing works and connections test successfully.

Solution: Verify that the clocks in the different RAC DB instances are in sync.

A

Using Multi Data Sources with Oracle RAC

Oracle recommends using GridLink data sources when developing new Oracle RAC applications. However, if you are using legacy applications and databases that do not support GridLink data sources, refer to the information in this appendix.

This appendix provides the following topics:

- Section A.1, "About Multi Data Sources and Oracle RAC"
- Section A.2, "Typical Procedure for Configuring Multi Data Sources for an EDG Topology"

A.1 About Multi Data Sources and Oracle RAC

A multi data source provides an ordered list of data sources to use to satisfy connection requests. Normally, every connection request to this kind of multi data source is served by the first data source in the list. If a database connection test fails and the connection cannot be replaced, or if the data source is suspended, a connection is sought sequentially from the next data source on the list.

For more information about configuring Multi Data Sources with Oracle RAC, see "Using Multi Data Sources with Oracle RAC" in the *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server*.

A.2 Typical Procedure for Configuring Multi Data Sources for an EDG Topology

You configure data sources when you configure a domain. For example, when you are configuring the initial Administration domain for an Enterprise Deployment reference topology, you use the configuration wizard to define the characteristics of the domain, as well as the data sources.

The procedures for configuring the topologies in this Enterprise Deployment Guide include specific instructions for defining GridLink data sources with Oracle RAC. If you want to use Multi Data Sources instead of GridLink data sources, replace the GridLink instructions with the following:

1. In the Configure JDBC Component Schema screen:

    a. Select the appropriate schemas.

    b. For the RAC configuration for component schemas, **Convert to RAC multi data source**.

    c. Ensure that the following data source appears on the screen with the schema prefix when you ran the Repository Creation Utility.

   d. Click **Next**.

   2. The Configure RAC Multi Data Sources Component Schema screen appears (Figure A–1).

Figure A–1   Configure RAC Multi Data Source Component Schema Screen
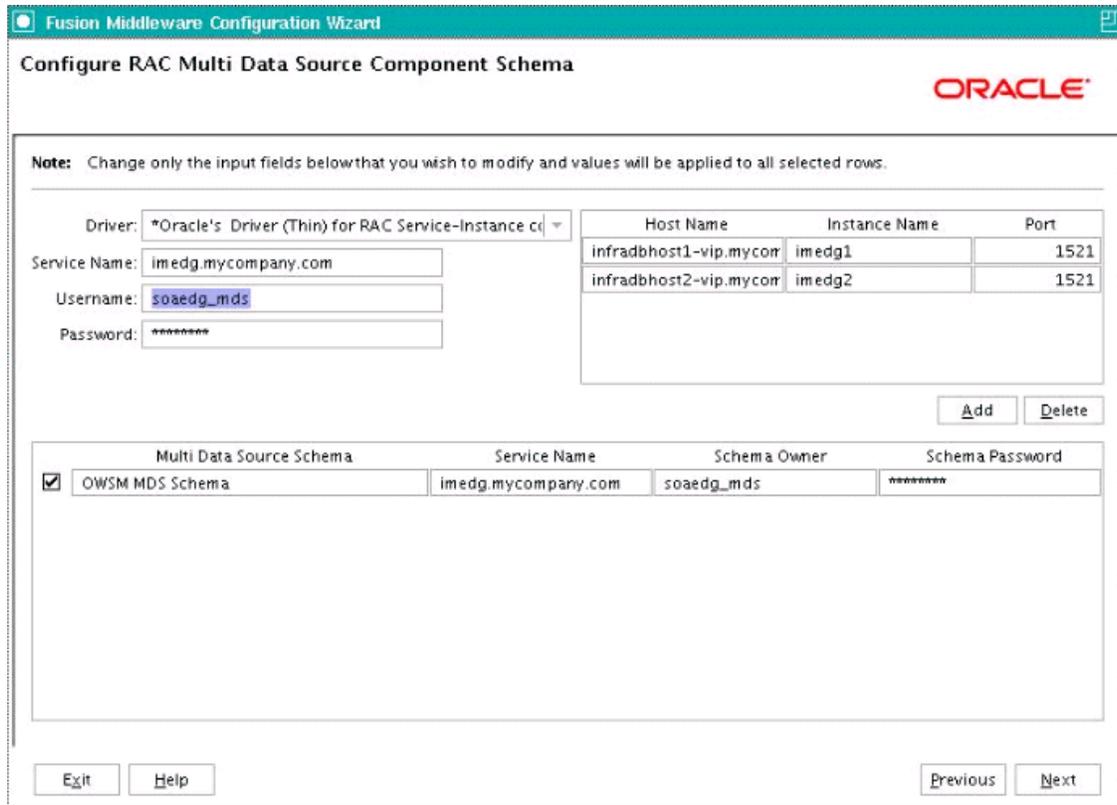
In this screen, do the following:

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

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

   – **Service Name:** Enter the service name of the database.

   – **Username:** Enter the complete user name (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**.

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

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

B

Targeting Applications and Resources to Servers

This appendix lists the applications, library, startup class, shutdown class, JMS system resource, WLDF system resource, and self-tuning resource targets for an Oracle SOA enterprise deployment.

This appendix includes the following sections:

- Section B.1, "Application Targets for Oracle SOA Suite"

- Section B.2, "Oracle SOA Enterprise Deployment Library Targets"

- Section B.3, "Oracle SOA Enterprise Deployment Startup Class Targets"

- Section B.4, "Oracle SOA Enterprise Deployment Shutdown Class Targets"

- Section B.5, "Oracle SOA Enterprise Deployment JMS System Resource Targets"

- Section B.6, "Oracle SOA Enterprise Deployment JDBC System Resource Targets"

- Section B.7, "Oracle SOA Enterprise Deployment WLDF system Resource Targets"

- Section B.8, "Oracle SOA Enterprise Deployment Self-Tuning Resource Targets"

B.1 Application Targets for Oracle SOA Suite

Table B–1 lists the Oracle SOA enterprise deployment application targets.

Table B–1   SOA Application Targets

Applications	Targets
FMW Welcome Page Application#11.1.0.0.0	AdminServer
DMS Application#11.1.1.1.0	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
wsil-wls	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
wsm-pm	WSM-PM_Cluster
em	AdminServer
NonJ2EEManagement#11.1.1	BAM_Cluster,AdminServer
FileAdapter	OSB_Cluster,AdminServer,SOA_Cluster
DbAdapter	OSB_Cluster,AdminServer,SOA_Cluster
JmsAdapter	OSB_Cluster,AdminServer,SOA_Cluster
AqAdapter	OSB_Cluster,AdminServer,SOA_Cluster

Table B–1   (Cont.) SOA Application Targets

Applications	Targets
FtpAdapter	OSB_Cluster,AdminServer,SOA_Cluster
SocketAdapter	OSB_Cluster,AdminServer,SOA_Cluster
MQSeriesAdapter	OSB_Cluster,AdminServer,SOA_Cluster
OracleAppsAdapter	OSB_Cluster,AdminServer,SOA_Cluster
OracleBamAdapter	OSB_Cluster,AdminServer,SOA_Cluster
usermessagingserver	BAM_Cluster,SOA_Cluster
usermessagingdriver-email	BAM_Cluster,SOA_Cluster
usermessagingdriver-xmpp	optional
usermessagingdriver-smpp	optional
usermessagingdriver-voicexml	optional
usermessagingdriver-extension	optional
UMSAdapter	OSB_Cluster,AdminServer,SOA_Cluster
soa-infra	SOA_Cluster
worklistapp	SOA_Cluster
b2bui	SOA_Cluster
DefaultToDoTaskFlow	SOA_Cluster
composer	SOA_Cluster
OracleBPMProcessRolesApp	SOA_Cluster
OracleBPMComposerRolesApp	SOA_Cluster
OracleBPMWorkspace	SOA_Cluster
BPMComposer	SOA_Cluster
SimpleApprovalTaskFlow	SOA_Cluster
composer	SOA_Cluster
OracleBPMProcessRolesApp	SOA_Cluster
OracleBPMComposerRolesApp	SOA_Cluster
OracleBPMWorkspace	SOA_Cluster
BPMComposer	SOA_Cluster
SimpleApprovalTaskFlow	SOA_Cluster
ALSB Cluster Singleton Marker Application	OSB_Cluster
ALSB Domain Singleton Marker Application	OSB_Cluster
ALSB Framework Starter Application	OSB_Cluster,AdminServer
ALSB Coherence Cache Provider	OSB_Cluster
XBus Kernel	OSB_Cluster,AdminServer
ALSB UDDI Manager	AdminServer
ALSB Subscription Listener	OSB_Cluster
JMS Reporting Provider	OSB_Cluster

Table B–1   (Cont.) SOA Application Targets

Applications	Targets
Message Reporting Purger	OSB_Cluster
Ftp Transport Provider	OSB_Cluster,AdminServer
SFTP Transport Provider	OSB_Cluster,AdminServer
Email Transport Provider	OSB_Cluster,AdminServer
File Transport Provider	OSB_Cluster,AdminServer
MQ Transport Provider	OSB_Cluster,AdminServer
EJB Transport Provider	OSB_Cluster,AdminServer
Tuxedo Transport Provider	OSB_Cluster,AdminServer
ALDSP Transport Provider	OSB_Cluster,AdminServer
SB Transport Provider	OSB_Cluster,AdminServer
WS Transport Provider	OSB_Cluster,AdminServer
WS Transport Async Applcation	OSB_Cluster
FLOW Transport Provider	OSB_Cluster,AdminServer
BPEL 10*g* Transport Provider	OSB_Cluster,AdminServer
JCA Transport Provider	OSB_Cluster,AdminServer
JEJB Transport Provider	OSB_Cluster,AdminServer
SOA-DIRECT Transport Provider	OSB_Cluster,AdminServer
ALSB Routing	OSB_Cluster,AdminServer
ALSB Transform	OSB_Cluster,AdminServer
ALSB Publish	OSB_Cluster,AdminServer
ALSB Logging	OSB_Cluster,AdminServer
ALSB Resource	OSB_Cluster
ALSB WSIL	OSB_Cluster
ServiceBus_Console	AdminServer
ALSB Test Framework	OSB_Cluster,AdminServer
oracle-bam#11.1.1	BAM_Cluster
ExternalPartnerSupplierEjb	SOA_Cluster
WebServices_WebLogicFusionOrderDemo_ CreditCardAuthorization	SOA_Cluster
ps5app	SOA_Cluster
FutureParticipantSimpleTaskForm	SOA_Cluster
AMXEDGTaskForm	SOA_Cluster
VacationRequestTaskFlow	SOA_Cluster
frevvo	SOA_Cluster
OWSM Policy Support in OSB Initializer Application	OSB_Cluster,AdminServer

B.2 Oracle SOA Enterprise Deployment Library Targets

Table B–2 lists the Oracle SOA enterprise deployment library targets.

Table B–2   Oracle SOA Enterprise Deployment Targeting Library

Library	Targets
oracle.bi.adf.view.slib#1.0@11.1.1.2.0	OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
oracle.bi.adf.webcenter.slib#1.0@11.1.1.2.0	OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
odl.clickhistory#1.0@11.1.1	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
odl.clickhistory.webapp#1.0@11.1.1	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
oracle.wsm.seedpolicies#11.1.1@11.1.1	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
oracle.jsp.next#11.1.1@11.1.1	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
oracle.dconfig-infra#11@11.1.1.1.0	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
orai18n-adf#11@11.1.1.1.0	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
oracle.adf.dconfigbeans#1.0@11.1.1.2.0	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
oracle.pwdgen#11.1.1@11.1.1.2.0	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
oracle.jrf.system.filter	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
adf.oracle.domain#1.0@11.1.1.2.0	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
adf.oracle.businesseditor#1.0@11.1.1.2.0	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
oracle.adf.management#1.0@11.1.1.2.0	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
adf.oracle.domain.webapp#1.0@11.1.1.2.0	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
jsf#1.2@1.2.9.0	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
UIX#11@11.1.1.1.0	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
ohw-rcf#5@5.0	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
ohw-uix#5@5.0	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
oracle.adf.desktopintegration.model#1.0@11.1.1.2.0	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
oracle.adf.desktopintegration#1.0@11.1.1.2.0	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster

Table B–2   (Cont.) Oracle SOA Enterprise Deployment Targeting Library

Library	Targets
oracle.bi.jbips#11.1.1@0.1	OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
oracle.bi.composer#11.1.1@0.1	OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
oracle.bpm.mgmt#11.1.1@11.1.1	SOA_Cluster
emai	AdminServer
oracle.webcenter.skin#11.1.1@11.1.1	AdminServer
oracle.webcenter.composer#11.1.1@11.1.1	AdminServer
emas	AdminServer
emcore	AdminServer
oracle.rules#11.1.1@11.1.1	BAM_Cluster,AdminServer,SOA_Cluster
oracle.sdp.client#11.1.1@11.1.1	BAM_Cluster,SOA_Cluster
oracle.soa.workflow.wc#11.1.1@11.1.1	SOA_Cluster
oracle.soa.worklist.webapp#11.1.1@11.1.1	SOA_Cluster
oracle.soa.rules_editor_dc.webapp#11.1.1@11.1.1	SOA_Cluster
oracle.soa.rules_dict_dc.webapp#11.1.1@11.1.1	BAM_Cluster,SOA_Cluster
oracle.sdp.messaging#11.1.1@11.1.1	SOA_Cluster,BAM_Cluster
oracle.soa.worklist#11.1.1@11.1.1	SOA_Cluster
oracle.soa.rules_editor_dc.webapp#11.1.1@11.1.1	SOA_Cluster
oracle.soa.rules_dict_dc.webapp#11.1.1@11.1.1	SOA_Cluster
oracle.soa.worklist#11.1.1@11.1.1	SOA_Cluster
oracle.soa.bpel#11.1.1@11.1.1	SOA_Cluster
oracle.soa.workflow#11.1.1@11.1.1	SOA_Cluster
oracle.soa.mediator#11.1.1@11.1.1	SOA_Cluster
oracle.soa.composer.webapp#11.1.1@11.1.1	SOA_Cluster
oracle.soa.ext#11.1.1@11.1.1	SOA_Cluster
oracle.bpm.client#11.1.1@11.1.1	SOA_Cluster
oracle.bpm.runtime#11.1.1@11.1.1	SOA_Cluster
oracle.bpm.projectlib#11.1.1@11.1.1	SOA_Cluster
oracle.bpm.webapp.common#11.1.1@11.1.1	SOA_Cluster
oracle.bpm.composerlib#11.1.1@11.1.1	SOA_Cluster
oracle.bpm.workspace#11.1.1@11.1.1	SOA_Cluster
oracle.jrf.coherence#3@11.1.1	OSB_Cluster,AdminServer
coherence-l10n#11.1.1@11.1.1	OSB_Cluster

Table B–2   (Cont.) Oracle SOA Enterprise Deployment Targeting Library

Library	Targets
ftptransport-l10n#2.5@2.5	OSB_Cluster,AdminServer
sftptransport-l10n#3.0@3.0	OSB_Cluster,AdminServer
emailtransport-l10n#2.5@2.5	OSB_Cluster,AdminServer
filetransport-l10n#2.5@2.5	OSB_Cluster,AdminServer
mqtransport-l10n#3.0@3.0	OSB_Cluster,AdminServer
mqconnection-l10n#3.0@3.0	OSB_Cluster,AdminServer
ejbtransport-l10n#2.5@2.5	OSB_Cluster,AdminServer
tuxedotransport-l10n#2.5@2.5	OSB_Cluster,AdminServer
aldsp_transport-l10n#3.0@3.0	OSB_Cluster,AdminServer
wstransport-l10n#2.6@2.6	OSB_Cluster,AdminServer
flow-transport-l10n#3.0@3.0	OSB_Cluster,AdminServer
bpel10gtransport-l10n#3.1@3.1	OSB_Cluster,AdminServer
jcatransport-l10n#3.1@3.1	OSB_Cluster,AdminServer
wsif#11.1.1@11.1.1	OSB_Cluster,AdminServer
JCAFrameworkImpl#11.1.1@11.1.1	OSB_Cluster,AdminServer
jejbtransport-l10n#3.2@3.2	OSB_Cluster,AdminServer
jejbtransport-jar#3.2@3.2	OSB_Cluster,AdminServer
soatransport-l10n#11.1.1.2.0@11.1.1.2.0	OSB_Cluster,AdminServer
stage-utils#2.5@2.5	OSB_Cluster,AdminServer
sbconsole-l10n#2.5@2.5	OSB_Cluster,AdminServer
xbusrouting-l10n#2.5@2.5	OSB_Cluster,AdminServer
xbustransform-l10n#2.5@2.5	OSB_Cluster,AdminServer
xbuspublish-l10n#2.5@2.5	OSB_Cluster,AdminServer
xbuslogging-l10n#2.5@2.5	OSB_Cluster,AdminServer
testfwk-l10n#2.5@2.5	OSB_Cluster,AdminServer
com.bea.wlp.lwpf.console.app#10.3.0@10.3.0	OSB_Cluster,AdminServer
com.bea.wlp.lwpf.console.web#10.3.0@10.3.0	OSB_Cluster,AdminServer
wlp-lookandfeel-web-lib#10.3.0@10.3.0	OSB_Cluster,AdminServer
wlp-light-web-lib#10.3.0@10.3.0	OSB_Cluster,AdminServer
wlp-framework-common-web-lib#10.3.0@10.3.0	OSB_Cluster,AdminServer
wlp-framework-struts-1.2-web-lib#10.3.0@10.3.0	OSB_Cluster,AdminServer
struts-1.2#1.2@1.2.9	OSB_Cluster,AdminServer
beehive-netui-1.0.1-10.0#1.0@1.0.2.2	OSB_Cluster,AdminServer
beehive-netui-resources-1.0.1-10.0#1.0@1.0.2.2	OSB_Cluster,AdminServer

Table B–2   (Cont.) Oracle SOA Enterprise Deployment Targeting Library

Library	Targets
beehive-controls-1.0.1-10.0-war#1.0@1.0.2.2	OSB_Cluster,AdminServer
weblogic-controls-10.0-war#10.0@10.2	OSB_Cluster,AdminServer
wls-commonslogging-bridge-war#1.0@1.1	OSB_Cluster,AdminServer
jstl#1.2@1.2.0.1	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
oracle.bam.library#11.1.1@11.1.1	BAM_Cluster
weblogic-sca#1.1@1.1.0.0	SOA_Cluster
oracle.bi.adf.model.slib#1.0@11.1.1.2.0	SOA_Cluster,BAM_Cluster,AdminServer

B.3 Oracle SOA Enterprise Deployment Startup Class Targets

Table B–3 lists Oracle SOA enterprise deployment startup class targets.

Table B–3   Oracle SOA Enterprise Deployment Startup Class Targets

Class	Targets
JRF Startup Class	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
JPS Startup Class	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
ODL-Startup	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
AWT Application Context Startup Class	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
JMX Framework Startup Class	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
Web Services Startup Class	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
JOC-Startup	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
DMS-Startup	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster
SOAStartupClass	SOA_Cluster
OSB JCA Transport Post-Activation Startup Class	OSB_Cluster,AdminServer

B.4 Oracle SOA Enterprise Deployment Shutdown Class Targets

Table B–4 lists Oracle SOA enterprise deployment shutdown classe targets.

Table B–4   SOA EDG targeting shutdown class

Class	Target
JOC-Shutdown	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster

Table B–4   (Cont.) SOA EDG targeting shutdown class

Class	Target
DMSShutdown	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster

B.5 Oracle SOA Enterprise Deployment JMS System Resource Targets

Table B–5 lists Oracle SOA enterprise deployment JMS resource targets.

Table B–5   JMS System Resource Targets

JMS Resource	Targets
UMSJMSSystemResource	BAM_Cluster,SOA_Cluster
PS6SOAJMSModule	SOA_Cluster
BPMJMSModule	SOA_Cluster
SOAJMSModule	SOA_Cluster
AGJMSModule	SOA_Cluster
WseeJmsModule	OSB_Cluster
jmsResources	OSB_Cluster
configwiz-jms	OSB_Cluster,BAM_Cluster
BAMJmsSystemResource	BAM_Cluster

B.6 Oracle SOA Enterprise Deployment JDBC System Resource Targets

Table B–6 lists Oracle SOA enterprise deployment JDBC system resource targets.

Table B–6   JDBC System Resource Targets

JMS Resource	Targets
mds-owsm	WSM-PM_Cluster,AdminServer
OraSDPMDataSource	BAM_Cluster,SOA_Cluster
SOADataSource	SOA_Cluster
EDNDataSource	SOA_Cluster
EDNLocalTxDataSource	SOA_Cluster
SOALocalTxDataSource	SOA_Cluster
mds-soa	AdminServer,SOA_Cluster
wlsbjmsrpDataSource	OSB_Cluster,AdminServer
leasing	OSB_Cluster,BAM_Cluster,SOA_Cluster
BAMDataSource	BAM_Cluster
jdbc/aqSample	SOA_Cluster
AppsAdapterTestDataSource	AdminServer,SOA_Cluster
RC4Connection	SOA_Cluster
oracle.bi.adf.model.slib	AdminServer,SOA_Cluster

B.7 Oracle SOA Enterprise Deployment WLDF system Resource Targets

Table B–7 lists Oracle SOA enterprise deployment WLDF resource targets.

Table B–7   WLDF System Resource Targets

JMS Resource	Targets
Module-FMWDFW	WSM-PM_Cluster,OSB_Cluster,BAM_Cluster,AdminServer,SOA_Cluster

B.8 Oracle SOA Enterprise Deployment Self-Tuning Resource Targets

Table B–8 lists Oracle SOA enterprise deployment self-tuning resource targets.

Table B–8   Self-Tuning Resource Targets

JMS Resource	Targets
wm/SOAWorkManager	SOA_Cluster
weblogic.wsee.mdb.DispatchPolicy	OSB_Cluster

Index