Oracle® Fusion Middleware

Enterprise Deployment Guide for Oracle Business Intelligence

11*g* Release 1 (11.1.1)

E15722-07

December 2014

Describes how to install and configure Oracle Business Intelligence components in an enterprise deployment.

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

E15722-07

Primary Authors: Kaye McArdell (Writer), Susan Kornberg (Contributing Engineer)

Contributing Authors: Edith Avot, Marla Azriel, Stuart Berry, Pradeep Bhat, Faouzia el-Idrissi, Jacek Laska, Yan Li, Srinivasan Varadarajan, Christine Jacobs

Contributor: High Availability Systems and Maximum Availability Architecture (MAA) and Oracle Business Intelligence development, product management, and quality assurance teams

Contents

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 with the Administration Server and First Managed Server

9 Scaling Out the Oracle Business Intelligence System

10 Setting Up Node Manager for an Enterprise Deployment

11 Configuring Server Migration for an Enterprise Deployment

12 Integrating an Enterprise Deployment with Oracle Identity Management

13 Managing Enterprise Deployments

Index

Preface

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

Audience

This document is intended for system administrators who are responsible for installing and configuring Oracle Business Intelligence enterprise deployments.

Documentation Accessibility

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

Access to Oracle Support

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

Related Documents

See the Oracle Business Intelligence documentation library for a list of related Oracle Business Intelligence documents.

See also the following related documents:

- *Oracle Fusion Middleware Release Notes* for your platform
- *Oracle Fusion Middleware High Availability Guide*
- *Oracle Fusion Middleware Administrator's Guide*
- *Oracle Fusion Middleware Repository Creation Utility User's Guide*
- *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory*

In addition:

- Go to the Oracle Learning Library for Oracle Business Intelligence-related online training resources.
- Go to the Product Information Center support note (Article ID 1267009.1) on My Oracle Support at `https://support.oracle.com`.

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 Business Intelligence.

Important: Oracle strongly recommends that you read the *Oracle Fusion Middleware Release Notes* for any additional installation and deployment considerations before starting the setup process.

This chapter contains the following sections:

- Section 1.1, "About the Enterprise Deployment Guide"
- Section 1.2, "Enterprise Deployment Terminology"
- Section 1.3, "Benefits of Oracle Recommendations"

1.1 About the Enterprise Deployment Guide

The Enterprise Deployment Guide defines an architectural blueprint that captures Oracle's recommended best practices for a highly available and secure Oracle Business Intelligence deployment. The best practices described in this blueprint use Oracle products from across the technology stack, including Oracle Database, Oracle Fusion Middleware, and Oracle Enterprise Manager Fusion Middleware Control. The resulting enterprise deployment can be readily scaled out to support increasing capacity requirements.

In particular, an Oracle Business Intelligence enterprise deployment:

- Considers various business service level agreements (SLAs) to make high-availability best practices as widely applicable as possible
- Leverages database grid servers and storage grids 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 that are independent of hardware and operating systems

For more information on high availability practices, see the Oracle Maximum Availability Architecture page on the Oracle Technology Network at:

http://www.oracle.com/technetwork/database/features/availability/maa-best-practices-155366.html

> **Note:** The Enterprise Deployment Guide for Oracle Business Intelligence 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 this guide.

- **Oracle home:** Contains installed files necessary to host a specific product. For example, the Oracle Business Intelligence Oracle home contains a directory that contains binary and library files for Oracle Business Intelligence. An Oracle home resides within the directory structure of the Middleware home. Each Oracle home can be associated with multiple Oracle instances or Oracle WebLogic Server domains.
- **WebLogic Server home:** Contains installed files necessary to host an Oracle WebLogic Server. The WebLogic Server home directory is a peer of the Oracle home directories and resides within the directory structure of the Middleware home.
- **Middleware home:** Consists of the 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:** Contains one or more active middleware system components, such as Oracle BI Server, Oracle BI Presentation Services, Oracle HTTP Server, or Oracle Internet Directory. You determine which components are part of an instance, either at installation time or by creating and configuring an instance at a later time. An Oracle instance contains files that can be updated, such as configuration files, log files, and temporary files.
- **failover:** The process that occurs when a member of a high availability system fails unexpectedly (unplanned downtime), so that the system can continue offering services to its consumers. If the system is an active-passive system, then the passive member is activated during the failover operation and consumers are directed to it instead of to the failed member. The failover process can be performed manually, or it can be automated by configuring 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, then the failover is performed by the load balancer entity serving requests to the active members. If an active member fails, then 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:** The process that occurs after a system undergoes a successful failover operation. In the failback process, the original failed member is repaired over time and is then reintroduced into the system as a standby member. Optionally, 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 collection of computers that provides a single view of network services (for example, an IP address) or application services (for example, databases and 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 with specialized hardware (cluster interconnect, shared storage) and software (health monitors, resource monitors). (The cluster interconnect is a private link used by the hardware cluster for heartbeat information to detect node death.) Due to the need for specialized hardware and software, hardware clusters are commonly provided by hardware vendors such as Sun, HP, IBM, and Dell. While the number of nodes that can be configured in a hardware cluster is vendor dependent, for the purpose of Oracle Fusion Middleware high availability, only two nodes are required. Hence, this document assumes a two-node hardware cluster for high availability solutions that employ a hardware cluster.

- **cluster agent:** The software that runs on a node member of a hardware cluster that coordinates availability and performance operations with other nodes. Clusterware provides resource grouping, monitoring, and the ability to move services. A cluster agent can automate the service failover.
- **clusterware:** Software that manages the operations of the members of a cluster as a system. It enables you to define a set of resources and services to monitor through a heartbeat mechanism between cluster members and to move these resources and services to a different member in the cluster as efficiently and transparently as possible.
- **shared storage:** The storage subsystem that is accessible by all the computers in the enterprise deployment. Among other things, the following is located on the shared disk:
  - Middleware home software
  - Administration Server 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, then the applicable Oracle Fusion Middleware components are failed over to the secondary node. This failover can be manual, or automated using the Clusterware for Administration Server. For a server migration-based scenario, WebLogic Whole Server Migration is used for automated failover.
- **secondary node:** The node that is the backup node for an Oracle Fusion Middleware instance. This is where the active instance fails over when the primary node is no longer available. See the definition for primary node in this section.
- **network host name:** A name assigned to an IP address either through the `/etc/hosts` file or through DNS resolution. This name is visible in the network where the computer to which it refers is connected. Often, the network host name and physical host name are identical. However, each computer has only one physical host name, but might have multiple network host names. Thus, a computer's network host name might not always be its physical host name.

- **physical host name:** The "internal name" of the current computer. On UNIX, this is the name returned by the hostname command. Note that this document differentiates between the terms physical host name and network host name.

  Oracle Fusion Middleware uses the physical host name to reference the local host. During installation, the installer automatically retrieves the physical host name from the current computer and stores it in the Oracle Fusion Middleware configuration metadata on disk.

- **physical IP:** The IP address of a computer on the network. In most cases, it is normally associated with the physical host name of the computer (see the definition for physical host name). In contrast to a virtual IP, it is always associated with the same computer when on a network.

- **switchover:** A process that occurs during normal operation when active members of a system might require maintenance or upgrading. A switchover process can be initiated to enable 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:** The process that occurs after a system undergoes a successful switchover operation, in which a member of the system is deactivated for maintenance or upgrading. When the maintenance or upgrade is completed, the system can undergo a switchback operation to activate the upgraded member and bring the system back to the pre-switchover configuration.

- **virtual host name:** A network addressable host name that maps to one or more physical computers through a load balancer or a hardware cluster. For load balancers, the name "virtual server name" is used interchangeably with virtual host name in this document. A load balancer can hold a virtual host name on behalf of a set of servers, and clients communicate indirectly with the computers 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 is explicitly stated.

- **virtual IP:** A virtual IP address that 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. Virtual IP is also called cluster virtual IP and load balancer virtual IP.

  A hardware cluster uses a cluster virtual IP to present to the outside world the entry point into the cluster (it can also be configured on a standalone computer). 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 computer has its own physical IP address and physical host name, while there might 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.

In addition to the terms defined in this section, this Enterprise Deployment Guide assumes knowledge of general Oracle Fusion Middleware and Oracle WebLogic Server concepts and architecture. See *Oracle Fusion Middleware Administrator's Guide* for more information.

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 Oracle Business Intelligence.

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 contains 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 demilitarized zone (DMZ), and all traffic is restricted by protocol and port. A DMZ is a perimeter network that exposes external services to a larger untrusted network.

The following characteristics ensure security at all needed levels and a high level of standards compliance:

- External load balancers are configured to redirect all external communication received on port 80 to port 443.

> **Note:** You can find a list of validated load balancers and their configuration on the Oracle Technology Network at:
>
> `http://www.oracle.com/technetwork/middleware/ias/tested-lbr-fw-sslaccel-100648.html`

- Communication from external clients does not go beyond the Load Balancing Router level.
- No direct communication from the Load Balancing Router to the data tier is allowed.
- Components are separated in different protection zones: the web tier, application tier, and the data tier.
- Direct communication between two firewalls at any one time is prohibited.
- If a communication begins in one firewall zone, then 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.

See also *Oracle Fusion Middleware High Availability Guide* for more information about high availability in Oracle Fusion Middleware.

2

Introduction to the Enterprise Deployment Reference Topology

This chapter describes and illustrates the enterprise deployment reference topology described in this guide. Use this chapter to help you plan the Oracle Business Intelligence enterprise deployment.

This chapter includes the following topics:

- Section 2.1, "Overview of Enterprise Deployment Reference Topologies"
- Section 2.2, "Hardware Requirements for an Enterprise Deployment on Linux"
- Section 2.3, "Identifying the Software Components to Install"
- Section 2.4, "About LDAP as Credential and Policy Store"
- Section 2.5, "Clock Synchronization"
- Section 2.6, "Road Map for the Reference Topology Installation and Configuration"

2.1 Overview of Enterprise Deployment Reference Topologies

The instructions and diagrams in this document describe a reference topology, to which variations might be applied. Use this section to plan the enterprise deployment topology.

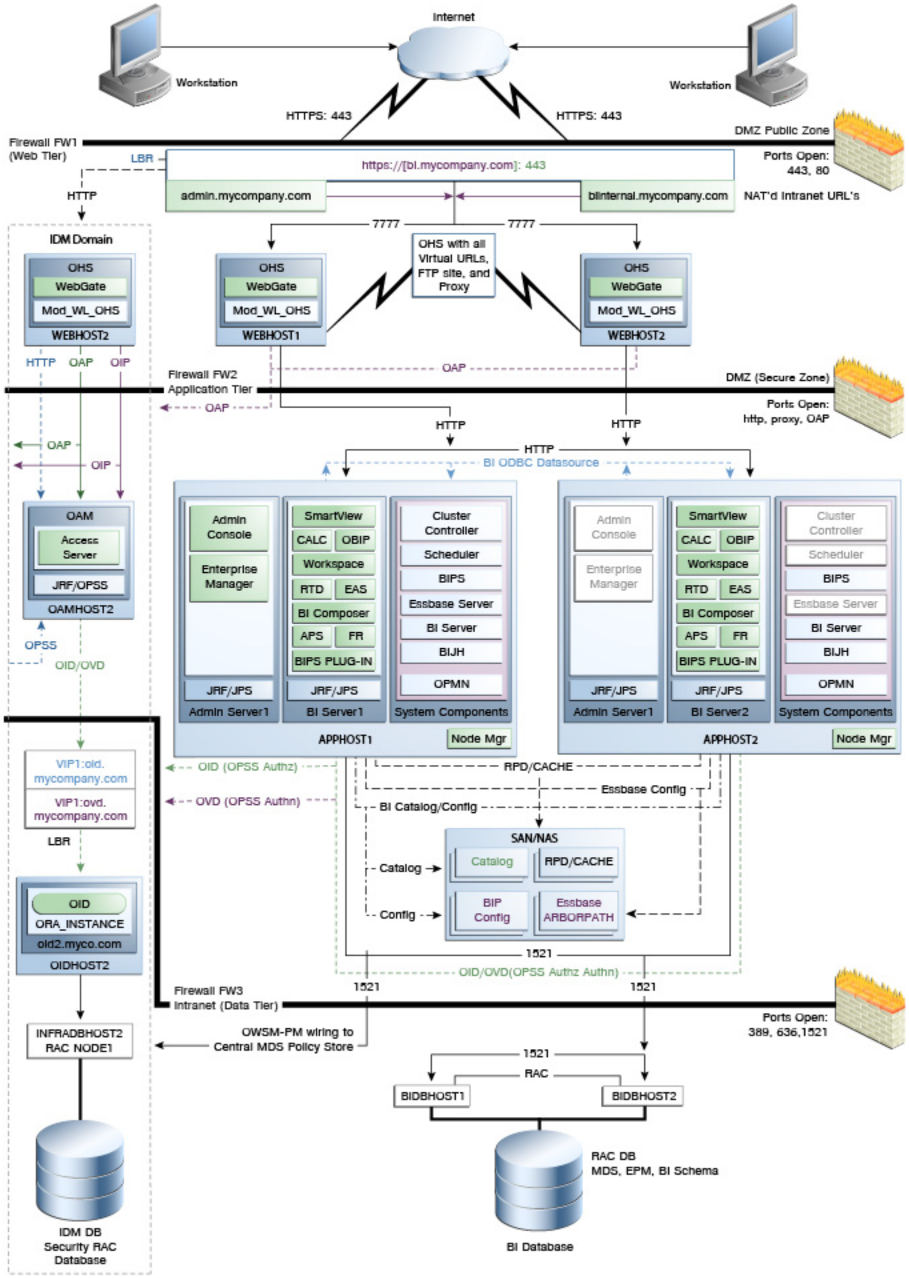
This section covers these topics:

- Section 2.1.1, "Reference Topologies Documented in the Guide"
- Section 2.1.2, "About Oracle Identity Management Integration"
- Section 2.1.3, "About the Web Tier Nodes"
- Section 2.1.4, "About the Application Tier"
- Section 2.1.5, "About the Data Tier"
- Section 2.1.6, "About the Unicast Requirement for Communication"

2.1.1 Reference Topologies Documented in the Guide

This document provides configuration instructions for a reference enterprise topology that uses Oracle Business Intelligence with Oracle Access Manager, as shown in Figure 2–1.

Figure 2–1 MyBICompany Topology with Oracle Access Manager



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 (Identity Management) 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, then requests 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 and 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, then 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 that is 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 that is 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 that protects the web tier, only the HTTP ports are open: 443 for HTTPS, and 80 for HTTP.

2.1.4 About the Application Tier

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

In this tier, APPHOST1 and APPHOST2 run the Oracle WebLogic Server Administration Console and Oracle Enterprise Manager Fusion Middleware Control, but in an active-passive configuration. You can fail over the Administration Server manually (see Section 13.5, "Manually Failing Over the Administration Server to APPHOST2"); alternatively, you can configure the Administration Console with CFC/CRS to fail over automatically on a separate hardware cluster (not shown in this architecture).

The Oracle Business Intelligence Cluster Controller, Oracle BI Scheduler, and Oracle Essbase Server system components run on APPHOST1 and APPHOST2 in an active-passive configuration. The other Oracle Business Intelligence system components, Oracle BI Server, Oracle BI JavaHost, and Oracle BI Presentation Services, run on APPHOST1 and APPHOST2 in an active-active configuration. All system components are managed by OPMN and do not run in the Managed Servers.

The Oracle Business Intelligence Java components, including Oracle Real-Time Decisions, Oracle BI Publisher, and the Oracle BI Presentation Services Plug-in (also called the Oracle BI Enterprise Edition Analytics application), run in the two Managed Servers on APPHOST1 and APPHOST2. Oracle Web Services Manager (Oracle WSM) is another Java component that provides a policy framework to manage and secure web services in the EDG topology. WSM Policy Manager runs in active-active configuration in the two Managed Servers in APPHOST1 and APPHOST2.

Starting with Release 11.1.1.7, Oracle BI Add-in for Microsoft Office is replaced by Oracle Hyperion Smart View for Office (Smart View) as a comprehensive tool for accessing and integrating Oracle Business Intelligence and Enterprise Performance Management content from Microsoft Office products. Smart View provides a common Microsoft Office interface designed for Oracle Business Intelligence and Oracle Enterprise Performance Management.

2.1.5 About the Data Tier

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

In this tier, an Oracle RAC database runs on the nodes CUSTDBHOST1 and CUSTDBHOST2. The database contains the schemas that the Oracle Business Intelligence components need. The Oracle Business Intelligence components that are running in the application tier access this database.

On the firewall that protects the data tier, the database listener port (typically, 1521) must be open. The LDAP ports (typically, 389 and 636) must also be open for the traffic that accesses the LDAP storage in the IDM EDG.

2.1.6 About the Unicast Requirement for Communication

Oracle recommends that the nodes in the MyBICompany 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 might not allow JMS multicast subscribers to work.
  - JMS multicast subscribers must be in a network hardware configuration that allows multicast accessibility. (That is, JMS subscribers must be in a multicast-enabled network to access multicast topics.)

2.2 Hardware Requirements for an Enterprise Deployment on Linux

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

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

You must perform the appropriate capacity planning to determine the number of nodes, CPU, and memory requirements for each node depending on the specific system's load, and the throughput and response requirements.

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
APPHOST*n*	20 GB or more	8 GB	Default	Default

2.3 Identifying the Software Components to Install

Table 2–2 lists the Oracle software that you must 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 11*g* or 12*c*	Oracle Database (in 11*g* series, 11.1.0.7 or higher)
Repository Creation Utility (RCU)	Oracle Fusion Middleware Repository Creation Utility 11*g* (11.1.1.9.0)
Oracle WebLogic Server (WLS)	Oracle WebLogic Server (10.3.6)
Oracle HTTP Server	Oracle Fusion Middleware WebTier and Utilities 11*g* (11.1.1.9.0)
Oracle Business Intelligence	Oracle Business Intelligence Enterprise Edition 11*g* (11.1.1.9.0)
Oracle Access Manager 10*g* Webgate *or* Oracle Access Manager 11*g* Webgate	Oracle Access Manager 10*g* Webgates (10.1.4.3.0); OAM OHS 11*g* Webgates per platform Oracle Access Manager 11*g* Webgates (11.1.1.5.0); OAM OHS 11*g* Webgates per platform
Oracle Virtual Directory (OVD)	Oracle Identity Management 11*g* (11.1.1.5.0)

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.

The Oracle Business Intelligence 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 store as the credential and policy store in the context of an Oracle Business Intelligence enterprise deployment. To configure the Oracle Business Intelligence enterprise deployment with an LDAP store as the credential and policy store, follow the steps in Section 12.1, "Configuring the Credential and Policy Store."

2.5 Clock Synchronization

The clocks of all servers that participate in the cluster must be synchronized to within one second difference to enable proper functioning of jobs and adapters. To accomplish this, use a single network time server and 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 the operating system documentation for more information.

2.6 Road Map for the Reference Topology Installation and Configuration

Table 2–3 describes each of the steps in the enterprise deployment process for Oracle Business Intelligence. The table also provides information on where to obtain more information on each step in the process.

Table 2–3 Steps in the Oracle Business Intelligence Enterprise Deployment Process

Step	Description	More Information
Prepare the Network for the Enterprise Deployment	Understand concepts like virtual server names, IPs, and virtual IPS, and configure the load balancer by defining virtual host names.	Chapter 3, "Preparing the Network for an Enterprise Deployment"
Prepare the File System for the 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 the Database for the Enterprise Deployment	Review database requirements, create database services, load the Oracle Business Intelligence schemas in the Oracle RAC database, and back up the database.	Chapter 5, "Preparing the Database for an Enterprise Deployment"
Install the Software for the Enterprise Deployment	Install Oracle HTTP Server, Oracle WebLogic Server, and Oracle Fusion Middleware.	Chapter 6, "Installing the Software for an Enterprise Deployment"
Configure the Web Tier for the Enterprise Deployment	Associate the Oracle web tier with the Oracle WebLogic Domain, configure the load balancer, and configure virtual host names.	Chapter 7, "Configuring the Web Tier for an Enterprise Deployment"
Create a Domain with the Administration Server and First Managed Server	Run the Oracle Business Intelligence Configuration Assistant to create a domain and perform post-configuration and verification tasks.	Chapter 8, "Creating a Domain with the Administration Server and First Managed Server"
Scale Out the Oracle Business Intelligence System	Extend the domain by scaling out Oracle Business Intelligence on APPHOST2, including scaling the BI system, scaling system components, configuring secondary instances of singleton system components, and performing additional high availability configuration.	Chapter 9, "Scaling Out the Oracle Business Intelligence System"
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 10, "Setting Up Node Manager for an Enterprise Deployment"
Configure Server Migration	Configure server migration for the bi_server1 and bi_server2 managed servers. The bi_server1 managed server is configured to restart on APPHOST2 if a failure occurs. The bi_server2 managed server is configured to restart on APPHOST1 if a failure occurs.	Chapter 11, "Configuring Server Migration for an Enterprise Deployment"
Integrate with Oracle Identity Management	Configure the credential and policy store, integrate with Oracle Access Manager 10*g* or 11*g*, and back up the Identity Management configuration.	Chapter 12, "Integrating an Enterprise Deployment with Oracle Identity Management"
Manage the Enterprise Deployment	Learn to start and stop Oracle Business Intelligence, monitor, scale, and patch the enterprise deployment, perform backups and recoveries, and review troubleshooting information.	Chapter 13, "Managing Enterprise Deployments"

3 Preparing the Network for an Enterprise Deployment

This chapter describes the network environment preconfiguration that the Oracle Business Intelligence enterprise topology requires. Use this chapter to plan the 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 Balancer"
- Section 3.4, "About IPs and Virtual IPs"
- Section 3.5, "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 must be configured to the appropriate real hosts and ports for the services that are running. Also, the load balancer must 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 must ensure that every computer where you plan to run Oracle Business Intelligence can access (ping) the primary host computer of the cluster using its fully qualified domain name. For the installation to succeed, every computer on which you scale out the installation must be able to support bidirectional communication with the Administration Server on the cluster's primary host.

3.2 About Virtual Server Names Used by the Topology

The BI enterprise topology uses the following virtual server names:

- bi.mycompany.com
- admin.mycompany.com
- biinternal.mycompany.com

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

In addition, the virtual IP addresses must be on the same subnet as the physical host IP addresses.

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

3.2.1 bi.mycompany.com

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

3.2.2 admin.mycompany.com

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

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

3.2.3 biinternal.mycompany.com

`biinternal.mycompany.com` is a virtual server name that is used for internal invocation of BI services. This URL is not exposed to the internet and is only accessible from the intranet.

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

3.3 Configuring the Load Balancer

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 that are 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.

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

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

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

3.3.1 Load Balancer Requirements

This enterprise topology uses 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

  This feature is necessary so that incoming requests on the virtual host name and port are directed to a different port on the back-end servers.

- Monitoring of ports on the servers in the pool to determine the availability of a service

- Ability to configure virtual server names and ports

  Note the following requirements:

  - The load balancer must allow configuration of multiple virtual servers. For each virtual server, the load balancer must allow configuration of traffic management on multiple ports. For example, for Oracle HTTP Server in the web tier, 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 the 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, then you should use it.

- Fault-tolerant mode

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

- Ability to configure the virtual server to return immediately to the calling client

  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 computer.

- Sticky routing capability

  Sticky routing capability is the ability to maintain sticky connections to components. Examples of this include cookie-based persistence, IP-based persistence, and so on.

- SSL acceleration

  The load balancer must have the ability to terminate SSL requests at the load balancer and forward traffic to the back-end real servers using the equivalent non-SSL protocol (for example, HTTPS to HTTP). Typically, this feature is called SSL acceleration and is required for this EDG.

- Connection timeout for TCP Connections

  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 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 Procedure

The procedure described in this section contains high-level steps. The actual steps that you perform vary depending on the type of load balancer that you use. For detailed instructions, consult the documentation for the load balancer.

Perform the following steps to configure the load balancer by defining the virtual server names:

1. Create a pool of servers. You assign this pool to virtual servers.
2. Add the addresses of the Oracle HTTP Server hosts to the pool. For example:
   - WEBHOST1:7777
   - WEBHOST2:7777
3. Configure a virtual server in the load balancer for `bi.mycompany.com:443` and define the following rules for this virtual server:
   - For this virtual server, use the system's frontend address as the virtual server address (for example, `bi.mycompany.com`). The frontend address is the externally facing host name that is used by the system that is exposed in the Internet.
   - Configure this virtual server with port 80 and port 443. Any request that goes to port 80 (non-ssl protocol) must be redirected to port 443 (ssl protocol).
   - Specify HTTP as the protocol.
   - Enable address and port translation.
   - Enable reset of connections when services and/or nodes are down.
   - Assign the pool created in Step 1 to the virtual server.
   - Create rules to filter out access to `/console` and `/em` on this virtual server.
4. Configure a virtual server in the load balancer for `admin.mycompany.com:80` and define the following rules for this virtual server:
   - For this virtual server, use the internal administration address as the virtual server address (for example, `admin.mycompany.com`). This address is typically not externalized.
   - Specify HTTP as the protocol.
   - Enable address and port translation.
   - Enable reset of connections when services and/or nodes are down.

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

5. Configure a virtual server in the load balancer for `biinternal.mycompany.com:80` and define the following rules for this virtual server:
   - For this virtual server, use the internal administration address as the virtual server address (for example, `biinternal.mycompany.com`). This address is typically not externalized.
   - Specify HTTP as the protocol.
   - Enable address and port translation.
   - Enable reset of connections when services and/or nodes are down.
   - Assign the pool created in Step 1 to the virtual server.
   - Optionally, create rules to filter out access to `/console` and `/em` on this virtual server.
6. Configure monitors for the Oracle HTTP Server nodes to detect failures in these nodes:
   - Configure a monitor to regularly ping the "/" URL context.

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

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

3.4 About IPs and Virtual IPs

Table 3–1 describes the various virtual hosts.

Table 3–1 Virtual Hosts

Virtual IP	VIP Maps to...	Description
VIP1	ADMINVHN	ADMINVHN is the virtual host name that is the listen address for the Administration Server and fails over with manual failover of the Administration Server. It is enabled on the node where the Administration Server process is running (APPHOST1 by default).
VIP2	APPHOST1VHN1	APPHOST1VHN1 is the virtual host name that maps to the listen address for bi_server1 and fails over with server migration of this Managed Server. It is enabled on the node where the bi_server1 process is running (APPHOST1 by default).
VIP3	APPHOST2VHN1	APPHOST2VHN1 is the virtual host name that maps to the listen address for bi_server2 and fails over with server migration of this Managed Server. It is enabled on the node where the bi_server2 process is running (APPHOST2 by default).

3.4.1 Enabling ADMINVHN on APPHOST1

The Administration Server must be configured to listen on a virtual IP Address to enable it to seamlessly failover from one host to another. In case of a failure, the Administration Server, along with the virtual IP Address, can be migrated from one host to another.

However, before the Administration Server can be configured to listen on a virtual IP Address, one of the network interface cards on the host that is running the Administration Server must be configured to listen on this virtual IP Address. The steps to enable a virtual IP Address are completely dependent on the operating system.

Perform the following steps to enable a virtual IP Address on APPHOST1. In a UNIX environment, the command must be run as the root user.

1. On APPHOST1, run the `ifconfig` command to get the value of the netmask. In a UNIX environment, run this command as the root user. For example:

```
[root@APPHOST1 ~] # /sbin/ifconfig
eth0     Link encap:Ethernet  HWaddr 00:11:43:D7:5B:06
     inet addr:139.185.140.51  Bcast:139.185.140.255  Mask:255.255.255.0
     inet6 addr: fe80::211:43ff:fed7:5b06/64 Scope:Link
     UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
     RX packets:10626133 errors:0 dropped:0 overruns:0 frame:0
     TX packets:10951629 errors:0 dropped:0 overruns:0 carrier:0
     collisions:0 txqueuelen:1000
     RX bytes:4036851474 (3.7 GiB)  TX bytes:2770209798 (2.5 GiB)
     Base address:0xecc0 Memory:dfae0000-dfb00000
```

2. On APPHOST1, bind the virtual IP Address to the network interface card using `ifconfig`. In a UNIX environment, run this command as the root user. Use a netmask value that was obtained in Step 1.

   The syntax and usage for the `ifconfig` command is as follows:

```
/sbin/ifconfig networkCardInterface Virtual_IP_Address netmask netMask
```

   For example:

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

3. Update the routing table using `arping`. In a UNIX environment, run this command as the root user.

```
/sbin/arping -q -U -c 3 -I networkCardInterface Virtual_IP_Address
```

   For example:

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

See also the following section for information about enabling VIP2 and VIP3 for the Managed Servers on APPHOST1 and APPHOST2.

3.4.2 Enabling Virtual IPs for the Managed Servers

The BI domain uses virtual host names as the listen addresses for the Oracle Business Intelligence Managed Servers. You must enable the VIPs, mapping each of these host names on the two Oracle BI computers (VIP2 on APPHOST1 and VIP3 on APPHOST2), and they must correctly resolve to the virtual host names in the network system that is used by the topology (either by DNS Server or hosts resolution).

Before the Managed Servers can be configured to listen on a virtual IP Address, one of the network interface cards on the host that is running the Managed Server must be configured to listen on this virtual IP Address.

Perform the following steps once on each host to enable the appropriate virtual IP Address (VIP2 on APPHOST1 and VIP3 on APPHOST2). In a UNIX environment, the command must be run as the root user:

1. On the appropriate host (APPHOST1 or APPHOST2), run the `ifconfig` command to get the value of the netmask. In a UNIX environment, run this command as the root user. For example, on APPHOST1:

```
[root@APPHOST1 ~] # /sbin/ifconfig
eth0     Link encap:Ethernet  HWaddr 00:11:43:D7:5B:06
     inet addr:139.185.140.51  Bcast:139.185.140.255  Mask:255.255.255.0
     inet6 addr: fe80::211:43ff:fed7:5b06/64 Scope:Link
     UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
     RX packets:10626133 errors:0 dropped:0 overruns:0 frame:0
     TX packets:10951629 errors:0 dropped:0 overruns:0 carrier:0
     collisions:0 txqueuelen:1000
     RX bytes:4036851474 (3.7 GiB)  TX bytes:2770209798 (2.5 GiB)
     Base address:0xecc0 Memory:dfae0000-dfb00000
```

2. Bind the virtual IP Address to the network interface card using `ifconfig`. In a UNIX environment, run this command as the root user. Use a netmask value that was obtained in Step 1.

   The syntax and usage for the `ifconfig` command is as follows:

```
/sbin/ifconfig networkCardInterface Virtual_IP_Address netmask netMask
```

   For example:

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

3. Update the routing table using `arping`. In a UNIX environment, run this command as the root user.

```
/sbin/arping -q -U -c 3 -I networkCardInterface Virtual_IP_Address
```

   For example:

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

See also Section 3.4.1, "Enabling ADMINVHN on APPHOST1" for information about enabling VIP1 for the Administration Server on APPHOST1.

3.5 About Firewalls and Ports

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

Most port numbers are assigned during installation.

Table 3–2 lists the ports that the Oracle Business Intelligence topology uses, including the ports that you must open on the firewalls in the topology.

Firewall notation:

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

- FW2 refers to the firewall between the application tier and the data tier.

Table 3–2 Ports Used

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Browser request	FW0	80	HTTP / Load Balancer	Inbound	Timeout depends on all HTML content and the type of process model used for BI.
Browser request	FW0	443	HTTPS / Load Balancer	Inbound	Timeout depends on all HTML content and the type of process model used for BI.
Load balancer to Oracle HTTP Server	n/a	7777	HTTP	n/a	See Section 3.3, "Configuring the Load Balancer."
Oracle HTTP Server registration with Administration Server	FW1	7001	HTTP/t3	Inbound	Set the timeout to a short period (5-10 seconds).
Oracle HTTP Server management by Administration Server	FW1	OPMN port (6701) and Oracle HTTP Server Admin Port (7779)	TCP and HTTP, respectively	Outbound	Set the timeout to a short period (5-10 seconds).
BI Server access	FW1	9704	HTTP/bi_server*n*	Inbound	Timeout varies based on the type of process model used for BI.
Communication between BI Cluster members	n/a	9704	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.
Oracle WebLogic Server Administration Console access	FW1	7001	HTTP / AdministrationServer and Enterprise Manager	Both	Tune this timeout based on the type of access to the Administration Console (whether you plan to use the Administration Console from application tier clients, or from clients external to the application tier).
Node Manager	n/a	5556	TCP/IP	n/a	n/a For actual values, see "Firewall and Port Configuration" in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*.

Table 3–2 (Cont.) Ports Used

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Access Server access	FW1	6021	OAP	Inbound	For actual values, see "Firewall and Port Configuration" in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*.
Identity Server access	FW1	6022	OAP	Inbound	n/a
Database access for BI Server and BI Publisher JDBC data sources	FW1	Listening port for client connections to the listener	SQL*Net	Both	Timeout depends on all database content and on the type of process model used for BI
Database access	FW2	1521	SQL*Net	Both	Timeout depends on all database content and on the type of process model used for BI.
Oracle Internet Directory access	FW2	389	LDAP	Inbound	Tune the directory server's parameters based on load balancer, and not the other way around.
Oracle Internet Directory access	FW2	636	LDAP SSL	Inbound	Tune the directory server's parameters based on load balancer, and not the other way around.
JOC for OWSM	n/a	9991	TCP/IP	n/a	n/a

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

4
Preparing the File System for an Enterprise Deployment

This chapter describes how to prepare the file system for an Oracle Business Intelligence 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:

4.1 Overview of Preparing the File System for Enterprise Deployment

It is important to set up the file system in a way that makes the enterprise deployment easier to understand, configure, and manage. Oracle recommends setting up the file 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 Variables

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

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

- **ORACLE_HOME:** This environment variable and related directory path refers to the directory where the binaries required to run Oracle Business Intelligence are installed.
- **ORACLE_COMMON_HOME:** This variable and related directory path refers to the Oracle home that contains the binary and library files required for Oracle Enterprise Manager Fusion Middleware Control and Java Required Files (JRF).
- **Domain directory:** This directory path refers to the location where the Oracle WebLogic domain information (configuration artifacts) is stored. Different WebLogic Servers can use different domain directories even when in the same node.
- **ORACLE_INSTANCE:** An Oracle instance directory contains configuration files, log files, and temporary files for one or more Oracle system components (such as Oracle BI Server, Oracle BI Presentation Services, Oracle HTTP Server, and so on).
- **JAVA_HOME:** This environment variable and related directory path refers to the location where JRockit is installed.
- **ASERVER_HOME:** This variable and related directory path refers the primary location of the domain configuration.
- **MSERVER_HOME:** This variable and related directory path refers to the domain configuration that is used to start and stop Managed Servers.
- **WEBGATE_ORACLE_HOME:** This variable and related directory path refers to 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 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 use %ORACLE_BASE% and use Windows-specific commands.

4.3 About Recommended Locations for the Different Directories

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

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

4.3.1 Shared Storage Recommendations for Binary (Oracle Home) Directories

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

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

4.3.1.1 About the Binary (Oracle Home) Directories

When you install any Oracle Fusion Middleware product, you install the product into an Oracle home. The binary files that are 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 is saved in a separate location from the domain configuration files, which you create using the Oracle Fusion Middleware Configuration Assistant.

The Middleware home for an Oracle Fusion Middleware installation contains the binary files 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 ability allows you to install the Oracle home in a single location on a shared volume and reuse the Oracle home for multiple host installations.

When an Oracle home is shared by multiple servers on different hosts, keep certain best practices in mind. In particular, ensure 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, enter the following command:

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

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

To update the Middleware home list to add or remove a MW_HOME, edit the `user_home/bea/beahomelist` file.

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 the 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.

If one Oracle home becomes corrupted or unavailable, then only half the servers are affected. For additional protection, Oracle recommends that you disk-mirror these volumes.

If separate volumes are not available on shared storage, then Oracle recommends simulating separate volumes using different directories within the same volume and mounting those to the same mount location on the host side. Although this approach does not guarantee the protection that multiple volumes provide, the approach 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 that you create when you configure Oracle Fusion Middleware products in an enterprise deployment:

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

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

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

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

In an enterprise deployment, bear in mind 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 one time.

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

Oracle recommends creating two copies of the domain configuration files:

- One copy is for the Administration Server configuration files.

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

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

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

  The Managed Servers' domain directories can reside in local or shared storage. Sharing domain directories for Managed Servers facilitates the scale-out procedures. However, sharing the Managed Server configuration files can also have a potential performance impact.

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

  The configuration steps that are 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 JTA Transaction Logs

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

For more information on using JMS and JTA information in a file store, see "Using the WebLogic Persistence Store" in *Oracle Fusion Middleware Configuring 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 when 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`*

- Mount point on system: *`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, then the mount point on the system is *`ORACLE_BASE`*`/admin/`*`domain_name`*`/mserver` mounted to *`ORACLE_BASE`*`/admin/`*`domain_name`*`/Node`*`n`*`/mserver/` (each node uses a different domain directory for Managed Servers).

> **Note:** This procedure is shared-storage dependent. The example in the preceding bullet point is specific to NAS, but other storage types might provide this redundancy with different types of mappings.

Location for JMS file-based stores and Tlogs:

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

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

- Mount point: *`ORACLE_BASE`*`/admin/`*`domain_name`*`/`*`cluster_name`*
- Shared storage location: *`ORACLE_BASE`*`/admin/`*`domain_name`*`/`*`cluster_name`*
- Mounted from: All nodes that run Oracle BI components must mount this shared storage location so that transaction logs and JMS stores are available when server migration to another node takes place.

Location for Application Directory for Administration Server:

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

- Mount point: *`ORACLE_BASE`*`/admin/`*`domain_name`*`/aserver`

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

Location for Application Directory for Managed Server:

`ORACLE_BASE/admin/domain_name/mserver/applications`

> **Note:** This directory is local in the context of the enterprise deployment for Oracle Business Intelligence.

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)
- Mounted from: Nodes alternatively mount VOL1 or VOL2 in such a way that at least half the nodes use an installation and the other half use the other one. In the enterprise deployment for Oracle Business Intelligence, APPHOST1 mounts VOL1 and APPHOST2 mounts VOL2. When only one volume is available, nodes mount two different directories in shared storage alternatively (that is, for example, APPHOST1 uses `ORACLE_BASE/product/fmw1` as shared storage location and APPHOST2 uses `ORACLE_BASE/product/fmw2` as shared storage location).

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

MW_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)
- Mounted from: For shared storage installations, nodes alternatively mount VOL1 or VOL2 in such a way that at least half the nodes use an installation and the other half use the other one. In the EDG for BI, WEBHOST1 mounts VOL1 and WEBHOST2 mounts VOL2. When only one volume is available, nodes mount the two suggested directories in shared storage alternatively (that is, for example, WEBHOST1 uses `ORACLE_BASE/product/fmw1` as shared storage location and WEBHOST2 uses `ORACLE_BASE/product/fmw2` as shared storage location).

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

WL_HOME:

Recommended directory = `MW_HOME/wlserver_10.3`

ORACLE_HOME:

Recommended directory = `MW_HOME/Oracle_BI1`

ORACLE_COMMON_HOME:

Recommended directory = `MW_HOME/oracle_common`

ORACLE_INSTANCE:

Recommended directory = `ORACLE_BASE/admin/instance_name`

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

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

Location for Oracle BI Presentation Catalog:

Recommended directory = `ORACLE_BASE/admin/domain_name/cluster_name/catalog/customCatalog` (where *customCatalog* is an example of the catalog name)

- Mount point: `ORACLE_BASE/admin/domain_name/cluster_name`
- Shared storage location: `ORACLE_BASE/admin/domain_name/cluster_name`
- Mounted from: All nodes that contain the instances of Presentation Services in the cluster mount this location (all nodes must have read and write access).

Note that the Oracle BI Presentation Catalog is called Presentation Service Repository in Fusion Middleware Control.

Location for Repository Publishing Directory:

Recommended directory = `ORACLE_BASE/admin/domain_name/cluster_name/ClusterRPD`

- Mount point: `ORACLE_BASE/admin/domain_name/cluster_name`
- Shared storage location: `ORACLE_BASE/admin/domain_name/cluster_name`
- Mounted from: All nodes that contain the instances of BI Server in the cluster mount this location. The master BI Server must have read and write access to this directory. All other BI Servers must have read access.

Note that the repository publishing directory is identified as the Shared Location for the BI Server Repository in Fusion Middleware Control.

Location for Essbase Agent Shared Folder Path

Recommended directory = `ORACLE_BASE/admin/domain_name/cluster_name/Essbase/essbaseserver1`

- Mount point: *ORACLE_BASE*/admin/*domain_name*/*cluster_name*
- Shared storage location: *ORACLE_BASE*/admin/*domain_name*/*cluster_name*
- Mounted from: All nodes that contain the instances of Essbase Agent in the cluster mount this location (all nodes must have read and write access).

Location for BI Server Global Cache:

Recommended directory = *ORACLE_BASE*/admin/*domain_name*/*cluster_name*/GlobalCache

- Mount point: *ORACLE_BASE*/admin/*domain_name*/*cluster_name*

  Shared storage location: *ORACLE_BASE*/admin/*domain_name*/*cluster_name*

  Mounted from: All nodes that contain the instances of BI Server in the cluster mount this location. The master BI Server must have read and write access to this directory. All other BI Servers must have read access.

Location for BI Publisher Configuration Folder:

Recommended directory = *ORACLE_BASE*/admin/*domain_name*/*cluster_name*/bipublisher/config

- Mount point: *ORACLE_BASE*/admin/*domain_name*/*cluster_name*
- Shared storage location: *ORACLE_BASE*/admin/*domain_name*/*cluster_name*
- Mounted from: All nodes that contain the instances of BI Publisher in the cluster mount this location with read/write access.

Location for BI Publisher Scheduler Temp Directory:

Recommended directory = *ORACLE_BASE*/admin/*domain_name*/*cluster_name*/bipublisher/temp

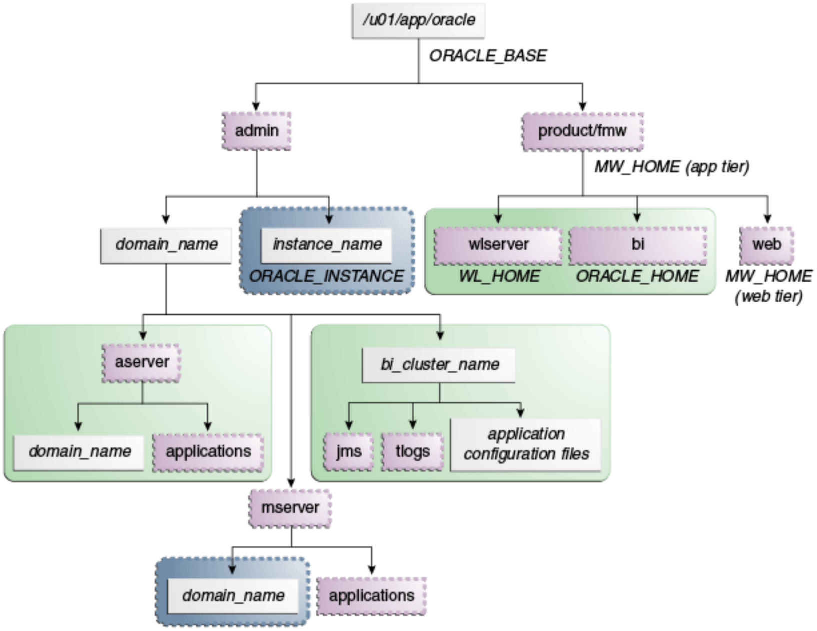
- Mount point: *ORACLE_BASE*/admin/*domain_name*/*cluster_name*
- Shared storage location: *ORACLE_BASE*/admin/*domain_name*/*cluster_name*
- Mounted from: All nodes that contain the instances of BI Publisher in the cluster mount this location with read/write access.

4.3.5 Directory Structure and Configurations

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

Figure 4–1 shows the recommended directory structure.

Figure 4–1 EDG Directory Structure for Oracle Business Intelligence



Note that the directory structure in Figure 4–1 does not show other required internal directories such as `oracle_common`. The structure shows directories for application configuration files such as the RPD Publishing directory, the shared Oracle BI Presentation Catalog, the Global Cache, and the Essbase Shared Folder Path. See Section 4.4, "Configuring Shared Storage" for information about shared directories.

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

Table 4–1 Directory Structure Elements

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

4.4 Configuring Shared Storage

Use the following commands to create and mount shared storage locations so that APPHOST1 and APPHOST2 can see the same location for binary installation in two separate volumes.

Note: The user ID that is 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 "Understanding Installation and Configuration Privileges and Users" section in *Oracle Fusion Middleware Installation Planning Guide*.

"nasfiler" is the shared storage filer.

From APPHOST1:

```
APPHOST1> mount nasfiler:/vol/vol1/u01/app/oracle/product/fmw
/u01/app/oracle/product/fmw -t nfs
```

From APPHOST2:

```
APPHOST2> mount nasfiler:/vol/vol2/u01/app/oracle/product/fmw
/u01/app/oracle/product/fmw -t nfs
```

If only one volume is available, then you can provide redundancy for the binary files by using two different directories in the shared storage and mounting them to the same directory in the APPHOST servers:

From APPHOST1:

```
APPHOST1> mount nasfiler:/vol/vol1/u01/app/oracle/product/fmw1
/u01/app/oracle/product/fmw -t nfs
```

From APPHOST2:

```
APPHOST2> mount nasfiler:/vol/vol2/u01/app/oracle/product/fmw2
/u01/app/oracle/product/fmw -t nfs
```

The following commands show how to share the location of the JTA transaction logs across different nodes:

```
APPHOST1> mount nasfiler:/vol/vol1/u01/app/oracle/stores/bifoundation_domain/
bi_cluster/tlogs /u01/app/oracle/admin/bifoundation_domain/
bi_cluster/tlogs -t nfs

APHOST2> mount nasfiler:/vol/vol1/u01/app/oracle/stores/bifoundatin_domain/
bi_cluster/tlogs /u01/app/oracle/admin/bifoundation_domain/
bi_cluster/tlogs -t nfs
```

Note: The shared storage can be a NAS or SAN device. The following illustrates an example of creating storage for a NAS device from APPHOST1. The options might differ depending on the specific storage device.

```
APPHOST1> 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 the storage vendor and computer administrator for the correct options for your environment.

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
```

4.4.1 Ensuring That Shared Network Files Are Accessible in Windows Environments

In Windows environments, shared storage is typically specified using Universal Naming Convention (UNC). UNC is a PC format for specifying locations of resources on a local area network. UNC uses the following format:

```
\\server_name\shared_resource_path_name
```

In addition, you must use named users to run OPMN processes in Windows environments so that the shared network files are accessible.

Perform the following steps to run OPMN processes using a named user:

1. Open the Services dialog. For example, select **Start > Programs > Administrative Tools > Services**.
2. Right-click **OracleProcessManager_instance*n*** and then select **Properties**.
3. Select the **Log On** tab.
4. Select **This account**, and provide a user name and password.
5. Click **OK**.

5

Preparing the Database for an Enterprise Deployment

This chapter describes procedures for preparing the database for an Oracle Business Intelligence enterprise deployment. The procedures include initial setup of the database, loading the Oracle Business Intelligence schemas, and backing up the database.

Important: **Oracle strongly recommends that you read the *Oracle Fusion Middleware Release Notes* for any additional installation and deployment considerations before starting the setup process.**

This chapter contains 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 for 11.x and 12c Databases"
- Section 5.4, "Loading the Oracle Business Intelligence Schemas in the Oracle RAC Database"
- Section 5.5, "Backing Up the Database"

5.1 Overview of Preparing the Database for an Enterprise Deployment

You must install a database and then load the Oracle Business Intelligence schemas into it before you can configure the Oracle Fusion Middleware components. You load the Oracle Business Intelligence schemas using the Repository Creation Utility (RCU).

For the enterprise topology, an Oracle Real Application Clusters (Oracle RAC) database is highly recommended to achieve a highly available data tier. When you install Oracle Business Intelligence, the installer prompts you to enter the information for connecting to the database that contains the required schemas.

5.2 About Database Requirements

Before loading the Oracle Business Intelligence schemas into the database, ensure that the database meets the requirements described in the following sections:

- Section 5.2.1, "Database Host Requirements"
- Section 5.2.2, "Supported Database Versions"

- Section 5.3, "Creating Database Services for 11.x and 12c Databases"
- Section 5.2.3, "Recommended Database Character Set"

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 2 (11.2) for Linux, refer to *Oracle Grid Infrastructure Installation Guide for Linux*. For 12*c* Release 1 (12.1) for Linux, refer to *Oracle Grid Infrastructure Installation Guide for Linux*.

- **Oracle Real Application Clusters**

  For 11*g* Release 2 (11.2) for Linux, refer to *Oracle Real Application Clusters Installation Guide for Linux and UNIX*. For 12*c* Release 1 (12.1), refer to *Oracle Real Application Clusters Installation Guide for Linux and UNIX*.

- **Automatic Storage Management (optional)**

  ASM is installed for the node as a whole. It is recommended that you install it in a separate Oracle home from the Oracle Database Oracle home. You can select this option when running the runInstaller. In the Select Configuration page, select the **Configure Automatic Storage Management** option to create a separate Oracle home for ASM.

5.2.2 Supported Database Versions

Oracle Business Intelligence requires the presence of a supported database and schemas.

To check if the 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 on the Oracle Technology Network at:

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

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

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

5.2.3 Recommended Database Character Set

Oracle strongly recommends using a database with AL32UTF8 as the database character set. You must select the AL32UTF8 character set when you install the database. If the database does not support AL32UTF8, then you see a warning when you run the Repository Creation Utility (RCU). Check the database documentation for information on choosing a character set for the database.

5.3 Creating Database Services for 11.*x* and 12*c* Databases

Oracle recommends that a specific database service be used for a product suite, even when product suites share the same database. It is also recommended that the database service used is different than the default database service. For complete instructions on creating database services, see the chapter on Workload Management in the *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide*.

Runtime connection load balancing requires configuring Oracle RAC Load Balancing Advisory with service-level goals for each service for which load balancing is enabled. The Oracle RAC Load Balancing Advisory can be configured for SERVICE_TIME or THROUGHPUT. Set the connection load balancing goal to SHORT. For 11*g* Release 1 Databases, use the DBMS_SERVICE package for this modification. For 11*g* Release 2 Databases, use the `srvctl` command utility instead.

This section contains the following topics:

- Section 5.3.1, "Creating Database Services for 11g Release 1 Databases"
- Section 5.3.2, "Creating Database Services for 11.2.x and 12c Databases"

5.3.1 Creating Database Services for 11*g* Release 1 Databases

The following steps provide an example for service creation and modification using the DBMS_SERVICE package:

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

```
prompt> sqlplus "sys/password as sysdba"
SQL> EXECUTE DBMS_SERVICE.CREATE_SERVICE
(SERVICE_NAME => 'biedg.mycompany.com',
NETWORK_NAME => 'biedg.mycompany.com'
);
```

Note: For the service name of the Oracle RAC database, use lowercase letters, followed by the domain name. For example, biedg.mycompany.com.

Also, the `EXECUTE DBMS_SERVICE` command shown must be entered on a single line to execute properly. 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`:

```
prompt> srvctl add service -d custdb -s biedg.mycompany.com -r custdb1,custdb2
```

3. Start the service using `srvctl`:

```
prompt> srvctl start service -d custdb -s biedg.mycompany.com
```

Note: For more information about the `SRVCTL` command, see 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 => 'biedg.mycompany.com',
goal => DBMS_SERVICE.GOAL_THROUGHPUT, clb_goal =>DBMS_SERVICE.CLB_GOAL_SHORT);
```

or:

```
SQL>EXECUTE DBMS_SERVICE.MODIFY_SERVICE (service_name => 'biedg.mycompany.com',
goal => DBMS_SERVICE.GOAL_SERVICE_TIME, clb_goal =>DBMS_SERVICE.CLB_GOAL_
SHORT);
```

5.3.2 Creating Database Services for 11.2.*x* and 12*c* Databases

The following steps provide an example for service creation and modification using the srvctl utility:

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

```
prompt> sqlplus "sys/password as sysdba"
SQL> EXECUTE DBMS_SERVICE.CREATE_SERVICE
(SERVICE_NAME => 'biedg.mycompany.com',
NETWORK_NAME => 'biedg.mycompany.com'
);
```

> **Note:** For the service name of the Oracle RAC database, use lowercase letters, followed by the domain name. For example, biedg.mycompany.com.
>
> Also, the `EXECUTE DBMS_SERVICE` command shown must be entered on a single line to execute properly. 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`:

```
prompt> srvctl add service -d custdb -s biedg.mycompany.com -r custdb1,custdb2
```

3. Start the service using `srvctl`:

```
prompt> srvctl start service -d custdb -s biedg.mycompany.com
```

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

4. Modify the service for the appropriate service goals:

```
prompt> srvctl modify service -d biedg -s biedg.mycompany.com -B
SERVICE_TIME -j SHORT
```

or:

```
prompt> srvctl modify service -d biedg -s biedg.mycompany.com -B
THROUGHPUT -j SHORT
```

5.4 Loading the Oracle Business Intelligence Schemas in the Oracle RAC Database

The Repository Creation Utility (RCU) is available from the RCU DVD. The RCU version used to seed the database must match the patch set level of the Oracle Business Intelligence installation. This means that if you install Oracle Business Intelligence 11*g* (11.1.1.9.0) in this enterprise deployment, you must use RCU 11*g* (11.1.1.9.0).

Perform the following steps to load the Oracle Business Intelligence schemas into the database:

1. Insert the Repository Creation Utility (RCU) DVD, and then start RCU from the bin directory in the RCU home directory.

```
prompt> cd RCU_HOME/bin
prompt> ./rcu
```

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

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

4. In the Database Connection Details screen, enter connect information for the database:

   - **Database Type**: Select **Oracle Database**.
   - **Host Name**: Specify the name of the node on which the database resides. For the Oracle RAC database, specify the VIP name or one of the node names as the host name: `CUSTDBHOST1-VIP.MYCOMPANY.COM`. For the SCAN-enabled RAC database, specify SCAN host as the host name.
   - **Port**: Specify the listen port number for the database: for example, *`1521`*.
   - **Service Name**: Specify the service name of the database (`biedg.mycompany.com`).
   - **Username**: Specify the name of the user with DBA or SYSDBA privileges: `SYS`.
   - **Password**: Enter the password for the SYS user.
   - **Role**: Select the database user's role from the list: `SYSDBA` (required by the SYS user).

   Click **Next**.

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

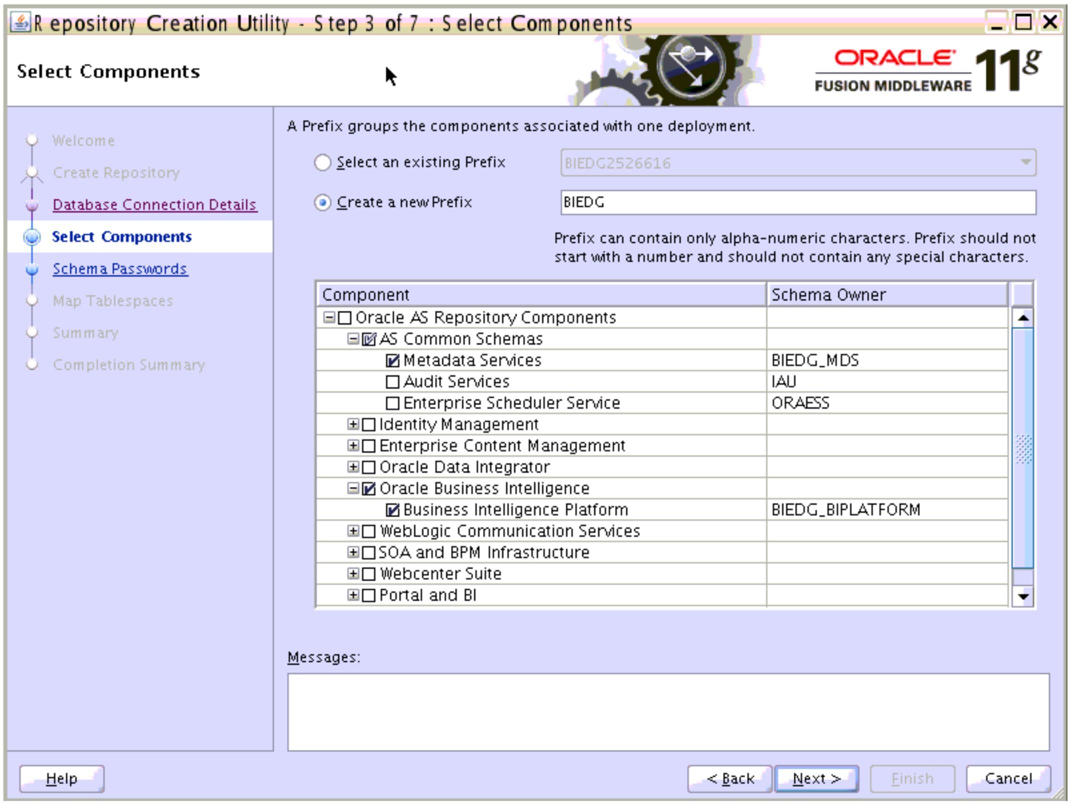
   - Select **Create a new Prefix**, and then enter a prefix to use for the database schemas (for example, `DEV` or `PROD`). You can specify as many as six characters as a prefix. Prefixes are used to create logical groupings of multiple repositories in a database. For more information, see the *Oracle Fusion Middleware Repository Creation Utility User's Guide*.

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

   - Select the following components:
     - **AS Common Schemas: Metadata Services** (automatically selected when **Oracle Business Intelligence: Business Intelligence Platform** is selected)
     - **Oracle Business Intelligence: Business Intelligence Platform**

   Click **Next**.

   The following image shows the Select Components screen:



Repository Creation Utility - Step 3 of 7 : Select Components

Select Components

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

A Prefix groups the components associated with one deployment.

○ Select an existing Prefix   BIEDG2526616

● Create a new Prefix   BIEDG

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

Component	Schema Owner
⊟☐ Oracle AS Repository Components	
⊟☒ AS Common Schemas	
☑ Metadata Services	BIEDG_MDS
☐ Audit Services	IAU
☐ Enterprise Scheduler Service	ORAESS
⊞☐ Identity Management	
⊞☐ Enterprise Content Management	
⊞☐ Oracle Data Integrator	
⊟☑ Oracle Business Intelligence	
☑ Business Intelligence Platform	BIEDG_BIPLATFORM
⊞☐ WebLogic Communication Services	
⊞☐ SOA and BPM Infrastructure	
⊞☐ Webcenter Suite	
⊞☐ Portal and BI	

Messages:

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

   You can choose either **Use same passwords for all schemas** or **Specify different passwords for all schemas**, depending on your requirements.

   Do not select **Use main schema passwords for auxiliary schemas**. The auxiliary passwords are derived from the passwords of the main schema users.

   > **Tip:** Note the names of the schema passwords, because the upcoming steps require this information.

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

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

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

About Oracle WSM Policies and the OWSM MDS Schemas

Oracle recommends using the database that is used for identity management to store the Oracle WSM policies. It is therefore expected that you use the identity management database information for the OWSM MDS schemas, which is different from the one that used for the rest of the Oracle BI schemas. To create the required schemas in the identity management database, repeat the preceding steps using the identity management database information, but select only **AS Common Schemas: Metadata Services** in the Select Components screen (step 5). See Chapter 12, "Integrating an Enterprise Deployment with Oracle Identity Management" for information on using the identity management database to store the Oracle WSM policies.

5.5 Backing Up the Database

After you have loaded the Oracle Business Intelligence schemas in the database, make a backup before installing the software for the enterprise deployment.

Backing up the database enables you to quickly recover from any issues that might occur in subsequent steps. You can choose to use the database backup strategy for this purpose, or you can simply make a backup using operating system tools or Oracle Recovery Manager (RMAN). Oracle recommends using RMAN 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.

6 Installing the Software for an Enterprise Deployment

This chapter describes the software installations that are required for the Oracle Business Intelligence enterprise deployment reference topology. You install Oracle HTTP Server and Oracle Fusion Middleware.

Important: Oracle strongly recommends that you read the *Oracle Fusion Middleware Release Notes* for any additional installation and deployment considerations before starting the setup process.

This chapter contains the following topics:

- 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 Oracle Fusion Middleware components. Later chapters describe the required configuration steps to create the reference topology for Oracle Business Intelligence.

Software to Install

Table 6–1 shows the software to install on each host or to 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 Business Intelligence
WEBHOST1	Yes	No	No
WEBHOST2	Yes	No	No
APPHOST1	No	Yes	Yes
APPHOST2	No	Yes	Yes

See Section 2.3, "Identifying the Software Components to Install" for information about software versions.

6.2 Installing Oracle HTTP Server

This section contains the following topics:

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

6.2.1 Prerequisites for Installing Oracle HTTP Server

Before installing Oracle HTTP Server, check that the computers meet the following requirements:

- Ensure that the system, patch, kernel, and other requirements are met as specified in *Oracle Fusion Middleware Installation Guide for Oracle Web Tier*.
- Because Oracle HTTP Server is installed on port 7777 by default, you must ensure that port 7777 is not used by any service on the nodes. To check if this port is in use, run the following command before installing Oracle HTTP Server:

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

  You must free port 7777 if it is in use.
- On Linux platforms, if the /etc/oraInst.loc file exists, then 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, then you can skip this step.
- Before starting the installation, ensure 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 install the web tier and you do *not* associate it with a domain. However, you must 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 the installation media:

   ```
   ./runInstaller
   ```

2. In the Specify Inventory Directory screen, perform the following steps:

   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 the screen to execute *HOME*`/orainventory/createCentralInventory.sh` as root, then click **OK**.

   The Specify Inventory Directory screen is displayed 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 that are installed on the machine.

3. In the Welcome screen, click **Next**.
4. In the Select Installation Type screen, select **Install Software - Do Not Configure**, and click **Next**.
5. In the Prerequisite Checks screen, verify that all checks complete successfully, and click **Next**.
6. In the Specify Installation Location screen, specify the following values:
   - **Oracle Middleware Home**: `ORACLE_BASE/product/fmw`
   - **Oracle Home Directory:** `Oracle_WT1`

   Note that these are two separate volumes on WEBHOST1 and WEBHOST2, even though the directory names are the same.

   Click **Next**.
7. In the Specify Security Updates screen, specify 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 that they are correct. If they are not, then 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, ensure that you run the script as the root user.

   The Oracle HTTP Server software is installed.
9. In the Installation Progress screen, click **Next**.
10. In the Installation Complete screen, click **Finish** to exit.

6.2.3 Backing Up the Oracle HTTP Server Installation

Back up the Middleware home for Oracle HTTP Server; ensure that you stop the server first:

```
WEBHOSTn> 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 Business Intelligence. The software components to install consist of the WebLogic Server home (*WL_HOME*) and Oracle home (*ORACLE_HOME*). As described in Chapter 4, "Preparing the File System for an Enterprise Deployment," you install Oracle Fusion Middleware in at least two storage locations for redundancy.

> **Important: Oracle strongly recommends that you read the *Oracle Fusion Middleware Release Notes* for any additional installation and deployment considerations before starting the setup process.**

This section covers the following topics:

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

- Section 6.3.2, "Installing Oracle Business Intelligence"
- Section 6.3.3, "Backing Up the Oracle Fusion Middleware Installation"

6.3.1 Installing Oracle WebLogic Server and Creating the Middleware Home

If you install Oracle WebLogic Server on 64-bit platforms using a 64-bit JDK, then follow the steps in "Installing WebLogic Server on 64-Bit Platforms Using a 64-Bit JDK" in *Oracle Fusion Middleware Installation Guide for Oracle WebLogic Server* instead of the steps in this section.

Perform these steps to install Oracle WebLogic Server on APPHOST1 and APPHOST2:

1. Start the installer for Oracle WebLogic Server from the installation media.
2. In the Welcome screen, click **Next**.
3. In the Choose Middleware Home Directory screen, do the following:
   - Select **Create a new Middleware Home**.
   - For Middleware Home Directory, enter `*ORACLE_BASE*/product/fmw`.

   > **Note:** *ORACLE_BASE* is the base directory under which Oracle products are installed. The recommended value is `/u01/app/oracle`. See Section 4.3, "About Recommended Locations for the Different Directories" for more information.
   >
   > Note that these are two separate volumes on APPHOST1 and APPHOST2, even though the directory names are the same.

   Click **Next**.
4. In the Register for Security Updates screen, specify whether you want to receive security updates from Oracle Support and if you do, enter your e-mail address.

   Click **Next**.
5. In the Choose Install Type screen, select **Typical** and click **Next**.
6. In the Choose Product Installation Directories screen, accept the directory ***ORACLE_BASE*/product/fmw/wlserver_10.3**. Also accept the default directory for Oracle Coherence (***ORACLE_BASE*/product/fmw/coherence_3.7**).
7. Click **Next**.
8. In the Installation Summary screen, click **Next**.

   The Oracle WebLogic Server software is installed.
9. In the Installation Complete screen, clear the **Run Quickstart** option and click **Done**.
10. Validate the installation by verifying that the following directories and files exist in the ORACLE_HOME directory after installing Oracle WebLogic Server:
    - `coherence_*version*`
    - `modules`
    - `registry.xml`
    - `utils`
    - `domain-registry.xml`

- `logs`
- `ocm.rsp`
- `registry.dat`
- `wlserver_10.3`

6.3.2 Installing Oracle Business Intelligence

Perform the following steps to install Oracle BI EE 11*g* (11.1.1.9.0) on APPHOST1 and APPHOST2. Note that because the installation is performed on a shared storage, the *MW_HOME* is accessible and used by the Oracle BI Servers in APPHOST1 and APPHOST2.

1. On Linux platforms, if the /etc/oraInst.loc file exists, verify 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, then you can skip this step.

2. Start the installer for Oracle Business Intelligence from the installation media:

   ```
   ./runInstaller
   ```

3. In the Specify Inventory Directory screen, perform the following steps:

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

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

   c. Click **Next**.

   Follow the instructions on screen to execute *HOME*`/orainventory/createCentralInventory.sh` as root, then click **OK**.

   The Specify Inventory Directory screen is displayed 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 that are installed on the computer.

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

5. In the Install Software Updates screen, specify whether to skip software updates, search My Oracle Support for software updates, or search the local directory for updates. When you are ready to proceed, click **Next**.

6. In the Select Installation Type screen, select **Software Only Install** and click **Next**.

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

8. In the Specify Installation Location screen, select the previously installed Middleware home from the drop-down list. For the Oracle home directory, enter the directory name (`Oracle_BI1`).

   Click **Next**.

9. In the Application Server screen, verify that **WebLogic Server** has been selected and click **Next**.

10. In the Specify Security Updates screen, specify whether you want to receive security updates from Oracle Support and if you do, enter your e-mail address and click **Next**.

11. In the Summary screen, click **Install**.

    The Oracle Business Intelligence software is installed.

12. In the Installation Progress screen, click **Next**.
13. In the Complete screen, click **Finish**.
14. Validate the installation by verifying that the following directories and files exist in the ORACLE_HOME directory after installing both Oracle WebLogic Server and Oracle Business Intelligence:

    - `coherence_`*`X.X`*
    - `jrockit-jdk`*`Y.Y`*
    - `modules`
    - `oracle_common`
    - `registry.xml`
    - `utils`
    - `domain-registry.xml`
    - `logs`
    - `ocm.rsp`
    - `registry.dat`
    - `Oracle_BI1`
    - `wlserver_10.3`

6.3.3 Backing Up the Oracle Fusion Middleware Installation

At this point, back up the Middleware home. Ensure that you stop the servers first.

To back up the Middleware home, run the following command:

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

This command creates a backup of the installation files for both Oracle WebLogic Server and the Oracle Fusion Middleware components, including Oracle Business Intelligence.

7

Configuring the Web Tier for an Enterprise Deployment

This chapter describes how to configure the Oracle Web Tier to support the Oracle Business Intelligence enterprise deployment.

Important: Oracle strongly recommends that you read the *Oracle Fusion Middleware Release Notes* for any additional installation and deployment considerations before starting the setup process.

This chapter contains the following topics:

7.1 Overview of Configuring the Web Tier

This chapter describes how to associate the Oracle Web Tier with the WebLogic Server domain. Once the Web Tier is associated with the WebLogic Server, you can monitor it using Oracle Enterprise Manager Fusion Middleware Control.

You then configure the load balancer to route all HTTP requests to WEBHOST1 and WEBHOST2.

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 Balancer."

7.2 Running the Configuration Wizard to Configure Oracle HTTP Server

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

Note: 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."

Perform the following steps to configure the Oracle Web tier:

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

   ```
   WEBHOSTn> cd ORACLE_BASE/product/fmw/Oracle_WT1/bin
   ```

2. Start the Configuration Wizard:

   ```
   WEBHOSTn> ./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**. Ensure that Oracle Web Cache is *not* selected.

   Click **Next**.

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

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

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

   Click **Next**.

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

6. In high-availability implementations, although it is not mandatory, it is simpler if all ports used by the various components are synchronized across hosts. You can bypass automatic port configuration by specifying the ports that you want to use in a file.

   In the Configure Ports screen, select a file name and click **View/Edit**. The file looks similar to the following:

   ```
   [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 in the `/Disk1/stage/Response/` directory.

   Click **Next**.

7. In the Specify Security Updates screen, specify 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 that they are correct. If they are not, then 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 the process completes, click **Next**, and the Installation Complete screen is displayed.

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

7.3 Validating the Configuration

After the configuration is complete, check that it is possible to access the Oracle HTTP Server home page using the following URLs:

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

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

7.4 Configuring the Load Balancer to Route HTTP Requests

Configure the load balancer to route all HTTP requests to the hosts that are 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 requests are routed directly from the load balancer to Oracle WebLogic Server, which is not the case in the topology described in this guide. Also, set monitors for HTTP.

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

7.5 Defining Virtual Hosts

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

This section contains the following topics:

- Section 7.5.1, "Defining the IP Address and Port in the httpd.conf File"
- Section 7.5.2, "Creating .conf Files to Define <VirtualHost> Directives"
- Section 7.5.3, "Validating the Configuration"

7.5.1 Defining the IP Address and Port in the httpd.conf File

You define name-based virtual servers. That means you must define the IP address and port to be used for each virtual host that 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.

You can find the httpd.conf file in the following directory:

```
ORACLE_INSTANCE/config/OHS/ohsn
```

To define the IP address and port, add the following entry in the httpd.conf file for both Oracle HTTP Servers:

```
NameVirtualHost *:7777
```

Note that the value for the NameVirtualHost parameter depends on how the virtual host for the load balancer was configured. You might need to use a different value than the one shown, such as 80.

7.5.2 Creating .conf Files to Define <VirtualHost> Directives

Define each virtual host in its own .conf file. This definition makes it easy to manage the URLs for each virtual host that you define.

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

- bi_vh.conf
- biinternal_vh.conf
- admin_vh.conf

Create the new files in the following directory:

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

> **Note:** ensure that you perform the following steps for both WEBHOST1 and WEBHOST2.

Perform the following steps to define each virtual host in its own .conf file:

1. Create the bi_vh.conf file and add the following directive:

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

2. Create the biinternal_vh.conf file and add the following directive:

   ```
   <VirtualHost *:7777>
      ServerName biinternal.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>
   ```

   If the steps in Chapter 12, "Integrating an Enterprise Deployment with Oracle Identity Management" have not been completed, then comment out the RewriteRule lines until the integration has been completed.

4. Restart both Oracle HTTP Servers, as follows:

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

7.5.3 Validating the Configuration

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

- https://bi.mycompany.com/index.html
- http://admin.mycompany.com/index.html
- http://biinternal.mycompany.com/index.html

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

8 Creating a Domain with the Administration Server and First Managed Server

This chapter describes how to create a domain with the Administration Server and the first Oracle Business Intelligence Managed Server using the Oracle Business Intelligence Configuration Assistant, Oracle WebLogic Server Administration Console, and Oracle Enterprise Manager Fusion Middleware Control. Later, you scale out the domain to add additional components. This is addressed in later chapters of this guide.

Important: Oracle strongly recommends that you read the *Oracle Fusion Middleware Release Notes* for any additional installation and deployment considerations before starting the setup process.

This chapter contains the following topics:

- Section 8.1, "Overview of Creating a Domain"
- Section 8.2, "Creating a Domain and the bi_server1 Managed Server on APPHOST1"
- Section 8.3, "Post-Configuration and Verification Tasks"
- Section 8.4, "Configuring Oracle HTTP Server for the WebLogic Domain"
- Section 8.5, "Verifying Manual Failover of the Administration Server"
- Section 8.6, "Backing Up the Installation"

8.1 Overview of Creating a Domain

Table 8–1 lists the steps for creating a WebLogic domain for Oracle Business Intelligence that contains the Administration Server and the first Managed Server, including post-configuration tasks.

Table 8–1 Steps for Creating a Domain for Oracle Business Intelligence

Step	Description	More Information
Create a WebLogic Domain	Run the Oracle Business Intelligence Configuration Assistant to create a WebLogic domain that contains the Administration Server and the first Managed Server.	Section 8.2, "Creating a Domain and the bi_server1 Managed Server on APPHOST1"
Post-Configuration and Verification Tasks	Follow the instructions for post-configuration and validation tasks.	Section 8.3, "Post-Configuration and Verification Tasks"
Configure Oracle HTTP Server for the WebLogic Domain	Configure Oracle HTTP Server for the Administration Server and Managed Server in the WebLogic domain and validate the configuration.	Section 8.4, "Configuring Oracle HTTP Server for the WebLogic Domain"
Test Administration Server Failover	Follow the instructions to manually fail over the Administration Server from APPHOST1 to APPHOST2.	Section 8.5, "Verifying Manual Failover of the Administration Server"
Back Up the Installation	Back up the installation to the local disk.	Section 8.6, "Backing Up the Installation"

8.2 Creating a Domain and the bi_server1 Managed Server on APPHOST1

Perform the following steps to run the Oracle Business Intelligence Configuration Assistant from the Oracle home directory to create a domain that contains the Administration Server and the first Managed Server with Oracle Business Intelligence components.

1. Ensure that the database where you installed the Business Intelligence Platform schemas is running. For an Oracle RAC database, it is recommended that you ensure that all instances are running, so that the validation check that is performed in later steps is more reliable.

2. Change the directory to the location of the Configuration Assistant (created in Chapter 6, "Installing the Software for an Enterprise Deployment"):

   ```
   APPHOST1> cd ORACLE_HOME/bin
   ```

3. Start the Configuration Assistant:

   ```
   APPHOST1> ./config.sh
   ```

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

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

6. The Create, Scale Out or Extend screen is displayed. In this screen, select **Create New BI System**, then enter the following:

   - **User Name:** `weblogic`
   - **User Password:** *`your_password`*
   - **Domain Name:** *`bifoundation_domain`*

   Click **Next**.

7. In the Specify Installation Location screen, enter:

   - **Middleware Home:** `ORACLE_BASE/product/fmw` (dimmed)
   - **Oracle Home:** `ORACLE_BASE/product/fmw/Oracle_BI1` (dimmed)

- **WebLogic Server Home:** `ORACLE_BASE/product/fmw/wlserver_10.3` (dimmed)
- **Domain Home:** `ORACLE_BASE/admin/domain_name/aserver/domain_name`

  The Domain Home must end with the domain name.
- **Instance Home:** `ORACLE_BASE/admin/instance1`
- **Instance Name:** `instance1`

Click **Next**.

8. In the Configure Components screen, select the following:

   - Oracle Business Intelligence
     - Business Intelligence Enterprise Edition
     - Business Intelligence Publisher
     - Real-Time Decisions
     - Essbase Suite

   Click **Next**.

9. In the BIPLATFORM Schema screen, provide the following information:

   - **Database Type:** `Oracle Database`
   - **Connect String:**

     ```
     CUSTDBHOST1-VIP.MYCOMPANY.COM:port-num:CUSTDB1^CUSTDBHOST2-VIP.MYCO
     MPANY.COM:port-num:CUSTDB2@BIEDG.MYCOMPANY.COM
     ```

     where *port-num* might be 1521.

     Note: For Oracle Real Application Cluster (RAC) 11*g* Release 2 and later, specify Single Client Access Name (SCAN). For earlier releases, specify the virtual IP address. Use a connect string such as the following one for SCAN:

     ```
     CUSTDB-SCAN.MYCOMPANY.COM:port-num:CUSTDB1^CUSTDB-SCAN.MYCOM
     PANY.COM:port-num:CUSTDB2@BIEDG.MYCOMPANY.COM
     ```

     Oracle strongly recommends that you read the *Oracle Fusion Middleware Release Notes* for any additional installation and deployment considerations for RAC support.

   - **BIPLATFORM Schema Username:** `prefix_BIPLATFORM`
   - **BIPLATFORM Schema Password:** *your_password*

   Click **Next**.

10. In the MDS Schema screen, verify the information. For example:

    - **Database Type:** `Oracle Database`
    - **Connect String:**

      ```
      CUSTDBHOST1-VIP.MYCOMPANY.COM:port-num:CUSTDB1^CUSTDBHOST2-VIP.MYCO
      MPANY.COM:port-num:CUSTDB2@BIEDG.MYCOMPANY.COM
      ```

      where *port-num* might be 1521.

> **Note:** For Oracle Real Application Cluster (RAC) 11*g* Release 2 and later, specify Single Client Access Name (SCAN). For earlier releases, specify the virtual IP address. Use a connect string such as the following one for SCAN:
>
> ```
> CUSTDB-SCAN.MYCOMPANY.COM:port-num:CUSTDB1^CUSTDB-SCAN.MYCOM
> PANY.COM:port-num:CUSTDB2@BIEDG.MYCOMPANY.COM
> ```
>
> Oracle strongly recommends that you read the *Oracle Fusion Middleware Release Notes* for any additional installation and deployment considerations for RAC support.

- **MDS Schema Username:** `prefix_MDS`
- **MDS Password:** *`your_password`*

Click **Next**.

11. In the Configure Ports screen, select one the following:

    - Auto Port Configuration
    - Specify Ports using Configuration File

    Click **Next**.

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

    Click **Next**.

13. In the Summary screen, click **Configure**.
14. In the Configuration Progress screen, verify that all the Configuration Tools have completed successfully and click **Next**.
15. In the Complete screen, click **Finish**.

Usually, Node Manager is started automatically when the config.sh process completes. If Node Manager is not running for some reason, then run these commands to start it on APPHOST1:

```
APPHOST1> cd WL_HOME/server/bin
APPHOST1> ./startNodeManager.sh
```

8.3 Post-Configuration and Verification Tasks

After configuring the domain with the Oracle Business Intelligence Configuration Assistant, follow these instructions for post-configuration and verification.

This section includes the following topics:

- Section 8.3.1, "Configuring JMS for BI Publisher"
- Section 8.3.2, "Creating boot.properties for the Administration Server on APPHOST1"
- Section 8.3.3, "Starting the Administration Server on APPHOST1"
- Section 8.3.4, "Enabling Administration Server High Availability"
- Section 8.3.5, "Validating the Administration Server"
- Section 8.3.6, "Setting the Listen Address for bi_server1 Managed Server"

- Section 8.3.7, "Disabling Host Name Verification for the bi_server1 Managed Server"
- Section 8.3.8, "Validating Oracle Business Intelligence on APPHOST1"

8.3.1 Configuring JMS for BI Publisher

Perform the following steps to configure the location for all persistence stores to a directory visible from both nodes. Change all persistent stores to use this shared base directory.

1. Log into the Administration Console.
2. In the Domain Structure window, expand the **Services** node and click the **Persistent Stores** node. The Summary of Persistent Stores page is displayed.
3. In the Change Center, click **Lock & Edit**.
4. Click **BipJmsStore** and enter a directory that is located in the shared storage. This shared storage is accessible from both APPHOST1 and APPHOST2:

   ```
   ORACLE_BASE/admin/domain_name/bi_cluster/jms
   ```

5. Click **Save** and **Activate Changes**.
6. The changes do not take effect until the Managed Server is restarted. Restart the Managed Server.

8.3.2 Creating boot.properties for the Administration Server on APPHOST1

Perform the following steps to create a boot.properties file for the Administration Server on APPHOST1. This file enables the Administration Server to start without prompting you for the administrator user name and password.

1. Go to the following directory:

   ```
   ORACLE_BASE/admin/domain_name/aserver/domain_name/servers/AdminServer/security
   ```

2. In this directory, create a file called boot.properties using a text editor and enter the following lines in the file:

   ```
   username=Admin_Username
   password=Admin_Password
   ```

   > **Note:** When you start the Administration Server, the user name and password entries in the file get encrypted. You start the Administration Server in Section 8.3.3, "Starting the Administration Server on APPHOST1." For security reasons, you want to minimize the time that the entries in the file are left unencrypted. After you edit the file, start the server as soon as possible so that the entries get encrypted.

3. Save the file and close the editor.

8.3.3 Starting the Administration Server on APPHOST1

Follow these steps to start the Administration Server using Node Manager:

1. Stop and restart Node Manager and enable dynamic registration, using the following steps:

a. Stop the running NodeManager process.

b. Restart Node Manager and enable dynamic registration using the following commands:

```
APPHOST1> cd WL_HOME/server/bin
APPHOST1> export JAVA_OPTIONS=-DDomainRegistrationEnabled=true
APPHOST1> ./startNodeManager.sh
```

> **Note:** It is important that you set the -DDomainRegistrationEnabled=true parameter whenever you start a Node Manager, which must manage the Administration Server. If there is no Administration Server on this computer and if this computer is not an Administration Server failover node, then you can start Node Manager as follows:
>
> ```
> APPHOST1> ./startNodeManager.sh
> ```
>
> If the computer on which the Administration Server is running is externally available, then you can choose not to set the -DDomainRegistrationEnabled=true parameter. If the -DDomainRegistrationEnabled parameter is not set to true, then you cannot start or stop the Administration Server using the Node Manager.

2. Start the Oracle WebLogic Scripting Tool (WLST) and connect to Node Manager with `nmconnect` and the credentials set in Step 1, and start the Administration Server using the `nmstart` command as follows:

```
APPHOST1> cd ORACLE_COMMON_HOME/common/bin
APPHOST1> ./wlst.sh
```

In the WLST shell, execute the following command:

```
wls:/offline>nmConnect('Admin_User','Admin_Pasword', 'APPHOST1',
'node_manager_port','domain_name','Domain_Home')

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

For example:

```
wls:/offline>nmConnect('weblogic', 'my_password', 'APPHOST1' , '9556',
'bifoundation_domain' ,'/u01/app/oracle/admin/bifoundation_domain/aserver/
bifoundation_domain')

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

> **Note:** APPHOST1 is the address of the node where the domain was created, not the listen address of the Administration Server. Also, the user name and password are used only to authenticate connections between Node Manager and clients. They are independent from the server admin ID and password and are stored in the `ORACLE_BASE/admin/domain_name/aserver/domain_name/config/nodemanager/nm_password.properties` file.

8.3.4 Enabling Administration Server High Availability

The Oracle WebLogic Server Administration Server is a singleton application, so it cannot be deployed in an active-active configuration. By default, the Administration Server is available only on the first installed node, and for this enterprise topology, it is available only on APPHOST1. If this node becomes unavailable, then the Administration Console and Fusion Middleware Control also become unavailable. To avoid this scenario, the Administration Server and the applications deployed to it must be enabled for high availability. The enterprise deployment architecture in this guide calls for the deploying the Administration Server on a disk shared between APPHOST1 and APPHOST2.

The process that is described in this guide initially deploys the Administration Server and the bi_server1 Managed Server on the shared disk that is mounted on APPHOST1, then manually migrates the bi_server1 Managed Server domain information to the local file system. This process is necessary to overcome certain design constraints in the Oracle Universal Installer.

> **Note:** Before performing the steps in the following sections, ensure that you have configured one of the network interface cards on the host that is running the Administration Server to listen on the virtual IP Address. See Section 3.4.1, "Enabling ADMINVHN on APPHOST1" for more information.

This section contains the following topics:

- Section 8.3.4.1, "Create a Machine for the Administration Server"
- Section 8.3.4.2, "Enabling the Administration Server to Listen on the Virtual IP Address"
- Section 8.3.4.3, "Creating a Separate Domain Directory for the bi_server1 Managed Server"
- Section 8.3.4.4, "Enabling Fusion Middleware Control Failover"

8.3.4.1 Create a Machine for the Administration Server

Perform the following steps to create a new machine and assign the Administration Server to the new machine using the Administration Console:

1. Log in to the Administration Console.
2. In the Change Center, click **Lock & Edit**.
3. In the Environment section of the Home page, click **Machines**.
4. On the Summary of Machines page, select the Machine that is associated with the Administration Server from under the Machines table and click **Clone**. For example: APPHOST1.MYCOMPANY.COM.
5. On the Clone a Machine page, enter the Name of the Machine under the Machine Identity section and click **OK**. For example, enter ADMINHOST as the machine name.
6. On the Summary of Machines page, click the newly created Machine link.
7. On the Settings page for the ADMINHOST machine, select the **Servers** tab.
8. Click **Add** under the Servers table.

9. On the Add a Server to Machine page, choose **Select an existing server, and associate it with this machine option**.
10. Choose the AdminServer from the drop-down menu.
11. Click **Finish** to associate the Administration Server with the Machine.
12. In the Change Center, click **Activate Changes**.

8.3.4.2 Enabling the Administration Server to Listen on the Virtual IP Address

Ensure that you have performed the steps described in Section 3.4.1, "Enabling ADMINVHN on APPHOST1" before setting the Administration Server listen address.

Perform the following steps to set the Administration Server listen address:

1. Log in to the Administration Console.
2. In the Change Center, click **Lock & Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page is displayed.
5. Select **AdminServer(admin)** in the Names column of the table. The Setting page for AdminServer(admin) is displayed.
6. Set the **Listen Address** to ADMINVHN.
7. Click **Save**.
8. Click **Activate Changes**.
9. The changes do not take effect until the Administration Server is restarted. Perform the following steps to restart the Administration Server:
   a. In the Summary of Servers page, select the **Control** tab.
   b. Select **AdminServer(admin)** in the table and then click **Shutdown**.
   c. Start the Administration Server again from the command line, as follows:

```
APPHOST1> cd ORACLE_COMMON_HOME/common/bin
APPHOST1> ./wlst.sh
wls:/offline> nmConnect
('Admin_User','Admin_Password','APPHOST1','node_mgr_port','domain_
name','Domain_home')
wls:/nm/domain_name> nmStart ('AdminServer')
```

8.3.4.3 Creating a Separate Domain Directory for the bi_server1 Managed Server

Perform the following steps to use the pack and unpack commands to separate the domain directory that is used by the Administration Server from the domain directory that is used by the bi_server1 Managed Server in APPHOST1:

1. Stop the Administration Server and bi_Server1 Managed Server, using the following steps:
   a. Log in to the Administration Console.
   b. Expand the **Environment** node in the Domain Structure window.
   c. Click **Servers**. The Summary of Servers page is displayed.
   d. On the Summary of Servers page, select the **Control** tab.
   e. Select **AdminServer(admin)** and **bi_server1** in the table and then click Shutdown.

2. Run the pack command on APPHOST1 to create a template pack using the following command. Ensure that you pass managed=true to pack just the bi_server1 Managed Server domain information.

   ```
   APPHOST1> cd ORACLE_HOME/common/bin
   APPHOST1> ./pack.sh -managed=true -domain=path_to_installer_created_domain
   -template=templateName.jar -template_name=templateName
   ```

   For example:

   ```
   APPHOST1> cd ORACLE_HOME/common/bin
   APPHOST1> ./pack.sh -managed=true -domain=
   ORACLE_BASE/admin/bifoundation_domain/aserver/bifoundation_domain
   -template=/tmp/managedServer.jar -template_name=ManagedServer_Template
   ```

3. Run the unpack command on APPHOST1 to unpack the template to the domain directory of the Managed Server using the following command:

   ```
   APPHOST1> cd ORACLE_HOME/common/bin
   APPHOST1> ./unpack.sh -domain=path_to_domain_on_LocalFileSystem
   -template=templateName.jar -app_dir=path_to_applications_dir_on_LocalFileSystem
   ```

   For example:

   ```
   APPHOST1> cd ORACLE_HOME/common/bin
   APPHOST1>./unpack.sh -domain=
   ORACLE_BASE/admin/bifoundation_domain/mserver/bifoundation_domain
   -template=/tmp/managedServer.jar -app_dir=
   ORACLE_BASE/admin/bifoundation_domain/mserver/applications
   ```

4. Start the Oracle WebLogic Scripting Tool (WLST) and connect to Node Manager with `nmconnect` and the credentials set in step 2, and start the Administration Server using `nmstart`:

   ```
   APPHOST1> cd ORACLE_COMMON_HOME/common/bin
   APPHOST1> ./wlst.sh
   ```

   In the WLST shell, execute the following command:

   ```
   wls:/offline>nmConnect('Admin_User','Admin_Pasword','APPHOST1',
   'node_manager_port','domain_name','Domain_Home')

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

   For example:

   ```
   wls:/offline>nmConnect('weblogic', 'my_password', 'APPHOST1' , '9556',
   'bifoundation_domain' ,'/u01/app/oracle/admin/bifoundation_domain/
   aserver/bifoundation_domain')

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

5. Start the bi_server1 Managed Server on APPHOST1, using the following steps (ensuring that Node Manager is up and running):

   a. Log in to the Administration Console.

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

   c. Click **Servers**. The Summary of Servers page is displayed.

   d. On the Summary of Servers page, select the **Control** tab.

   e. Select **bi_server1** in the table and then click **Start**.

8.3.4.4 Enabling Fusion Middleware Control Failover

To enable Fusion Middleware Control failover, copy the em.ear file from the *MW_HOME*/user_projects/applications/bifoundation_domain directory to the equivalent directory on all nodes where high availability for Administration Server might be performed. In some cases, you might need to create the *MW_HOME*/user_projects/applications/bifoundation_domain directory on the other nodes.

8.3.5 Validating the Administration Server

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

1. Open a Web browser and go to http://ADMINVHN:7001/console.
2. Log in as the administrator.
3. Check that you can access Fusion Middleware Control at:

   http://ADMINVHN:7001/em
4. Log in to Fusion Middleware Control with the user name and password that you specified in Section 8.3.2, "Creating boot.properties for the Administration Server on APPHOST1."

8.3.6 Setting the Listen Address for bi_server1 Managed Server

Ensure that you have performed the steps that are described in Section 3.4.2, "Enabling Virtual IPs for the Managed Servers" before setting the bi_server1 listen address.

Perform the following steps to set the listen address for the Managed Server:

1. Log in to the Administration Console.
2. In the Change Center, click **Lock & Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page is displayed.
5. Select **bi_server1** in the Names column of the table. The Setting page for bi_server1 is displayed.
6. Set the **Listen Address** to APPHOST1VHN1.
7. Click **Save**.
8. Click **Activate Changes**.
9. The changes do not take effect until the bi_server1 Managed Server is restarted using the following steps (ensure that Node Manager is up and running):
   1. On the Summary of Servers page, select the **Control** tab.
   2. Select **bi_server1** in the table and then click **Shutdown**.
   3. After the server has shut down, select **bi_server1** in the table and then click **Start**.
10. Restart the Oracle Business Intelligence system components, as follows:

```
cd ORACLE_BASE/admin/instance1/bin
./opmnctl stopall
./opmnctl startall
```

8.3.6.1 Updating the Logical Web Applications in the EPM Registry

Perform the following steps to update the host information for the Logical Web Applications endpoints of EPM applications in the EPM registry:

1. Enter the following command to obtain the GUID for the logical web application for Oracle Business Intelligence:

   ```
   /u01/app/oracle/admin/instance1/config/foundation/11.1.2.0/epmsys_
   registry.sh view LOGICAL_WEB_APP
   ```

   Multiple components are returned by the command.

2. Change all the HOST values to the internal load balancer virtual host for each of the components. Update the port to 80:

   ```
   /u01/app/oracle/admin/instance1/config/foundation/11.1.2.0/epmsys_
   registry.sh updateproperty \#<Component_GUID>/@host
   biinternal.mycompany.com
   ```

   ```
   /u01/app/oracle/admin/instance1/config/foundation/11.1.2.0/epmsys_
   registry.sh updateproperty \#<Component_GUID>/@port 80
   ```

   where *GUID* is the ID of the EPM application in the EPM registry.

3. Verify that the following properties have the appropriate values:

   - HOST = The internal LBR vip. For example, biinternal.mycompany.com.
   - isSSL = False.
   - port = 80.

4. Restart the Managed Servers and BI System components.

8.3.7 Disabling Host Name Verification for the bi_server1 Managed Server

This step is required if you have not configured the appropriate certificates to authenticate the different nodes with the Administration Server (see Chapter 10, "Setting Up Node Manager for an Enterprise Deployment"). If you have not configured the server certificates, then you see errors when managing the different Oracle WebLogic Servers. To avoid these errors, disable host name verification while configuring and validating the topology, and enable it again after the EDG topology configuration is complete, as described in Chapter 10, "Setting Up Node Manager for an Enterprise Deployment."

Perform the following steps to disable host name verification:

1. Log in to the Administration Console.
2. In the Change Center, click **Lock & Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page is displayed.
5. Select **bi_server1** in the Names column of the table. The settings page for the server is displayed.
6. Open the **SSL** tab.
7. Expand the **Advanced** section of the page.
8. Set **Hostname Verification** to 'None'.
9. Click **Save**.

10. Click **Activate Changes**.
11. The change does not take effect until the bi_server1 Managed Server is restarted using the following steps (ensure that Node Manager is up and running):
    a. On the Summary of Servers page, select the **Control** tab.
    b. Select **bi_server1** in the table and then click **Shutdown**.
    c. Select **bi_server1** in the table and then click **Start**.
12. Restart the Oracle Business Intelligence system components, as follows:

```
cd ORACLE_BASE/admin/instance1/bin
./opmnctl stopall
./opmnctl startall
```

8.3.8 Validating Oracle Business Intelligence on APPHOST1

Access the following URLs:

- Access http://APPHOST1VHN1:9704/analytics to verify the status of bi_server1.
- Access http://APPHOST1VHN1:9704/wsm-pm to verify the status of Web Services Manager. Click **Validate Policy Manager**. A list of policies and assertion templates available in the data is displayed.

  Note: The configuration is incorrect if no policies or assertion templates appear.
- Access http://APPHOST1VHN1:9704/xmlpserver to verify the status of the BI Publisher application.
- Access http://APPHOST1VHN1:9704/ui to verify the status of the Oracle RTD application.
- Access http://APPHOST1VHN1:9704/bicomposer to verify the status of the Oracle BI Composer application.
- Access http://APPHOST1VHN1:9704/aps/Essbase to verify the status of the Oracle Essbase application.
- Access http://APPHOST1VHN1:9704/aps/SmartView to verify the status of the Smart View application.
- Access http://APPHOST1VHN1:9704/workspace to verify the status of the Workspace application.
- Access http://APPHOST1VHN1:9704/hr to verify the status of the Financial Reporting application.
- Access http://APPHOST1VHN1:9704/calcmgr/index.htm to verify the status of the Calculation Manager application.

8.4 Configuring Oracle HTTP Server for the WebLogic Domain

This section describes how to configure Oracle HTTP Server for the WebLogic domain that contains the Administration Server and the bi_server1 Managed Server.

This section contains the following topics:

- Section 8.4.1, "Configuring Oracle HTTP Server for the Administration Server and the bi_servern Managed Servers"
- Section 8.4.2, "Turning On the WebLogic Plug-In Enabled Flag"
- Section 8.4.3, "Registering Oracle HTTP Server with Oracle WebLogic Server"

- Section 8.4.4, "Setting the Frontend URL for the Administration Console"
- Section 8.4.5, "Validating Access Through Oracle HTTP Server"

8.4.1 Configuring Oracle HTTP Server for the Administration Server and the bi_server*n* Managed Servers

To enable Oracle HTTP Server to route to the Administration Server and the bi_cluster, which contains the bi_server*n* Managed Servers, you must set the WebLogicCluster parameter to the list of nodes in the cluster.

Perform the following steps to set the WebLogicCluster parameter:

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

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

   Note that this step assumes you created the admin_vh.conf and biinternal_vh.conf files using the instructions in Section 7.5, "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>
   ```

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

   ```
   #redirect browser requests that omit document/dir
   RedirectMatch 301 /analytics$ /analytics/
   RedirectMatch 301 /bimiddleware$ /bimiddleware/
   RedirectMatch 301 /analytics/res$ /analytics/res/
   RedirectMatch 301 /bioffice$ /bioffice/
   RedirectMatch 301 /biofficeclient$ /biofficeclient/
   RedirectMatch 301 /biservices$ /biservices/
   RedirectMatch 301 /analytics-ws$ /analytics-ws/
   RedirectMatch 301 /AdminService$ /AdminSErvice/
   RedirectMatch 301 /AsyncAdminService$ /AsyncAdminService/
   RedirectMatch 301 /rtis$ /rtis/
   RedirectMatch 301 /wsm-pm$ /wsm-pm/
   RedirectMatch 301 /xmlpserver$ /xmlpserver/
   RedirectMatch 301 /ui$ /ui/
   RedirectMatch 301 /bisearch$ /bisearch/
   RedirectMatch 301 /mapviewer$ /mapviewer/
   ```

```
RedirectMatch 301 /bicontent$ /bicontent/
RedirectMatch 301 /bicomposer$ /bicomposer/
RedirectMatch 301 /mobile$ /mobile/
RedirectMatch 301 /rtis$ /rtis/
RedirectMatch 301 /aps$ /aps/
RedirectMatch 301 /calcmgr$ /calcmgr/
RedirectMatch 301 /hr$ /hr/
RedirectMatch 301 /workspace$ /workspace/

# WSM-PM
<Location /wsm-pm>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

# BIEE Analytics
<Location /analytics>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

<Location /analytics-ws>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

<Location /bimiddleware>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

<Location /bicontent>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

<Location /mobile>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

# MapViewer
<Location /mapviewer>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

# BI Publisher
<Location /xmlpserver>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

# Oracle RTD
<Location /ui>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

<Location /rtis>
```

```
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

<Location /schema>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

<Location /ws>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

# BI Search
<Location /bisearch>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

# BI Composer

<Location /bicomposer>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

# EPM Provider Services
<Location /aps>
   SetHandler weblogic-handler
   WeblogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

# EPM Calc Manager
<Location /calcmgr>
   SetHandler weblogic-handler
   WeblogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

# EPM Financial Reporting
<Location /hr>
   SetHandler weblogic-handler
   WeblogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

# EPM Workspace
<Location /workspace>
   SetHandler weblogic-handler
   WeblogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

# BI Office
<Location /bioffice>
   SetHandler weblogic-handler
   WeblogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

# BI Office Client
<Location /biofficeclient>
   SetHandler weblogic-handler
   WeblogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

# BI SOA Services
<Location /biservices>
   SetHandler weblogic-handler
   WeblogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

# AdminService
<Location /AdminService>
   SetHandler weblogic-handler
   WeblogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

# AsyncAdminService
<Location /AsyncAdminService>
   SetHandler weblogic-handler
   WeblogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>

#OWSM
<Location /wsm-pm>
   SetHandler weblogic-handler
   WeblogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
</Location>
```

3. Add the following directives to the bi_vh.conf file within the <VirtualHost> tags:

```
#redirect browser requests that omit document/dir
RedirectMatch 301 /analytics$ /analytics/
RedirectMatch 301 /bimiddleware$ /bimiddleware/
RedirectMatch 301 /xmlpserver$ /xmlpserver/
RedirectMatch 301 /ui$ /ui/
RedirectMatch 301 /analytics/res$ /analytics/res/
RedirectMatch 301 /biofficeclient$ /biofficeclient/
RedirectMatch 301 /biservices$ /biservices/
RedirectMatch 301 /analytics-ws$ /analytics-ws/
RedirectMatch 301 /ws$ /ws/
RedirectMatch 301 /wsm-pm$ /wsm-pm/
RedirectMatch 301 /bisearch$ /bisearch/
RedirectMatch 301 /mapviewer$ /mapviewer/
RedirectMatch 301 /bicontent$ /bicontent/
RedirectMatch 301 /bicomposer$ /bicomposer/
RedirectMatch 301 /mobile$ /mobile/
RedirectMatch 301 /rtis$ /rtis/
RedirectMatch 301 /aps$ /aps/
RedirectMatch 301 /calcmgr$ /calcmgr/
RedirectMatch 301 /hr$ /hr/
RedirectMatch 301 /workspace$ /workspace/

# BIEE Analytics
<Location /analytics>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>

<Location /analytics-ws>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>

<Location /bimiddleware>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>

<Location /bicontent>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>

<Location /mobile>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>

# MapViewer
<Location /mapviewer>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>

# BI Publisher
<Location /xmlpserver>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>

# Oracle RTD
<Location /ui>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>

<Location /rtis>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>

<Location /schema>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>

<Location /ws>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>

# BI Search
<Location /bisearch>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>

# BI Composer

<Location /bicomposer>
   SetHandler weblogic-handler
   WebLogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>

# EPM Provider Services
<Location /aps>
   SetHandler weblogic-handler
   WeblogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>

# EPM Calc Manager
<Location /calcmgr>
   SetHandler weblogic-handler
   WeblogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>

# EPM Financial Reporting
<Location /hr>
   SetHandler weblogic-handler
   WeblogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>

# EPM Workspace
<Location /workspace>
   SetHandler weblogic-handler
   WeblogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>
```

```
# BI Office
<Location /bioffice>
   SetHandler weblogic-handler
   WeblogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>

# BI Office Client
<Location /biofficeclient>

   SetHandler weblogic-handler
   WeblogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>

# OWSM
<Location /wsm-pm>
   SetHandler weblogic-handler
   WeblogicCluster APPHOST1VHN1:9704,APPHOST2VHN1:9704
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>
```

Note: Add other resources as appropriate (such as analyticsRes or ActionSamples to support functionality in the SampleApp.rpd file).

4. 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
```

The servers that are specified in the WebLogicCluster parameters are important only at startup time for the plug-in. The list must provide at least one running cluster member for the plug-in to discover other members in 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 include:

- **Example 1:** If you have a two-node cluster and add a third member, then you do not need to update the configuration to add the third member. The third member is discovered dynamically 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 fails 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 the members of the cluster, then you guarantee that 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 *Oracle Fusion Middleware Using Web Server Plug-Ins with Oracle WebLogic Server*.

8.4.2 Turning On the WebLogic Plug-In Enabled Flag

For security purposes, and because the load balancer terminates SSL requests (Oracle HTTP Server routes the requests as non-SSL to WebLogic Server), after SSL is configured for the load balancer, turn on the WebLogic Plugin Enabled flag for the domain. Perform the following steps to turn on the flag:

1. Log in to the Administration Console.
2. Click the domain name in the navigation tree on the left.
3. Click the **Web Applications** tab.
4. In the Change Center, click **Lock & Edit**.
5. Select **WebLogic Plugin Enabled**.
6. Click **Save**, then click **Activate Changes**.

8.4.3 Registering Oracle HTTP Server with Oracle WebLogic Server

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

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

You must also run this command from WEBHOST2 for OHS2.

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

8.4.4 Setting the Frontend URL for the Administration Console

The 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, then the console redirects the HTTP request, which replaces the host and port information with the listen address and port of the Administration Server. When the Administration Console is accessed using a load balancing router (LBR), you must change the frontend URL of the Administration Server so that the user's web browser is redirected to the appropriate LBR address. Perform the following steps to set the frontend URL:

1. Log in to the Administration Console.
2. In the Change Center, click **Lock & Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page is displayed.
5. Select **AdminServer(admin)** in the Names column of the table. The settings page for AdminServer(admin) is displayed.

6. Click the **Protocols** tab.
7. Click the **HTTP** tab.
8. Set the **Frontend Host** field to `admin.mycompany.com` (your LBR address).
9. Set the **Frontend HTTP Port** to 80.
10. Click **Save**, then click **Activate Changes**.

> **Note:** To eliminate redirections, it is recommended that you disable the Administration Console's "Follow changes" feature. Ensure that the **Follow Configuration Changes** option in the Shared Preferences dialog under the Preferences menu is not selected.

8.4.5 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," then wait for the server status to change to "Started." If another status is reported (such as "Admin" or "Failed"), then check the server output log files for errors.

This section contains the following topics:

- Section 8.4.5.1, "Validating the Administration Console and Fusion Middleware Control"
- Section 8.4.5.2, "Validating Access Through the Load Balancer"

8.4.5.1 Validating the Administration Console and Fusion Middleware Control

Validate the Administration Console and Fusion Middleware Control through both Oracle HTTP Server instances using the following URLs:

- http://WEBHOST*n*:7777/console
- http://WEBHOST*n*:7777/em

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

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

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

8.4.5.2 Validating Access Through the Load Balancer

Validate bi_cluster through the load balancer using the following URLs:

- http://bi.mycompany.com/analytics
- http://bi.mycompany.com/mapviewer
- http://bi.mycompany.com/xmlpserver
- http://bi.mycompany.com/ui

- http://biinternal.mycompany.com/wsm-pm
- http://bi.mycompany.com/bicomposer
- http://bi.mycompany.com/hr
- http://bi.mycompany.com/calcmgr/index.htm
- http://bi.mycompany.com/aps/Essbase
- http://bi.mycompany.com/aps/SmartView
- http://bi.mycompany.com/workspace

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

8.5 Verifying Manual Failover of the Administration Server

In case a node fails, you can fail over the Administration Server to another node. Test Administration Server failover at this point by following the steps in Section 13.5, "Manually Failing Over the Administration Server to APPHOST2."

8.6 Backing Up the Installation

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

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

1. Back up the Web tier, using the following steps:

   a. Shut down the instance using `opmnctl`:

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

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

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

   c. Back up the Oracle instance on the Web tier using the following command:

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

   d. Start the instance using `opmnctl`:

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

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

3. Back up the BI Instance in the application tier, using the following steps:

   a. Shut down the instance using opmnctl:

      ```
      APPHOST1> ORACLE_INSTANCE/bin/opmnctl stopall
      ```

   b. Back up the Middleware home on the application tier using the following command:

      ```
      APPHOST1> tar -cvpf BACKUP_LOCATION/bi.tar MW_HOME
      ```

   c. Back up the Oracle instance on the application tier using the following command:

      ```
      APPHOST1> tar -cvpf BACKUP_LOCATION/bi_instance_name.tar ORACLE_INSTANCE
      ```

   d. Start the instance using opmnctl:

      ```
      APPHOST1> ORACLE_INSTANCE/bin/opmnctl startall
      ```

4. Back up the Administration Server and Managed Server domain directories to save the domain configuration. The configuration files all exist in the `ORACLE_BASE/admin/ domain_name` directory. Run the following command to create the backup:

   ```
   APPHOST1> tar -cvpf edgdomainback.tar ORACLE_BASE/admin/domain_name
   ```

9 Scaling Out the Oracle Business Intelligence System

This chapter describes how to scale out the Oracle Business Intelligence system using the Configuration Assistant. It is assumed that an Oracle Business Intelligence *ORACLE_HOME* (binaries) has already been installed and is available from APPHOST1 and APPHOST2, and that a domain with an Administration Server has been created. This is the domain that you extend in this chapter to support Oracle Business Intelligence components.

> **Important:** **Oracle strongly recommends that you read the *Oracle Fusion Middleware Release Notes* for any additional installation and deployment considerations before starting the setup process.**

This chapter contains the following topics:

- Section 9.1, "Overview of Scaling Out the Oracle Business Intelligence System"
- Section 9.2, "Setting Up Shared File Locations"
- Section 9.3, "Scaling Out the Oracle Business Intelligence System on APPHOST2"
- Section 9.4, "Configuring the bi_server2 Managed Server"
- Section 9.5, "Performing Additional Configuration for Oracle Business Intelligence Availability"
- Section 9.6, "Other Post-Configuration and Verification Tasks"
- Section 9.7, "Backing Up the Installation"

9.1 Overview of Scaling Out the Oracle Business Intelligence System

Table 9–1 lists and describes the steps to scale out the Oracle Business Intelligence system and perform post-configuration tasks.

Table 9–1 Steps for Scaling Out the Oracle Business Intelligence System

Step	Description	More Information
Set Up Shared File Locations	Set up locations for shared files in Oracle BI EE and BI Publisher.	Section 9.2, "Setting Up Shared File Locations"
Scale Out the Oracle Business Intelligence System	Use the Oracle Business Intelligence Configuration Assistant to scale out the Oracle Business Intelligence system, then use Fusion Middleware Control to scale out system components and configure secondary instances for singleton system components.	Section 9.3, "Scaling Out the Oracle Business Intelligence System on APPHOST2"
Configure the bi_server2 Managed Server	Set the listen address and disable host name verification for the bi_server2 Managed Server.	Section 9.4, "Configuring the bi_server2 Managed Server"
Perform Additional Configuration for Oracle Business Intelligence Availability	Perform additional high availability configuration tasks for Oracle BI Scheduler, Oracle RTD, BI Publisher, and Oracle BI Composer.	Section 9.5, "Performing Additional Configuration for Oracle Business Intelligence Availability"
Perform Other Post-Configuration and Verification Tasks	Configure a default persistence store for transaction recovery, start and validate Oracle Business Intelligence on APPHOST2, and configure Node Manager and server migration for the Managed Servers.	Section 9.6, "Other Post-Configuration and Verification Tasks"
Back Up the Installation	Back up the installation to the local disk.	Section 9.7, "Backing Up the Installation"

9.2 Setting Up Shared File Locations

This section describes how to set up locations for shared files in Oracle BI EE and BI Publisher. It contains the following topics:

- Section 9.2.1, "Setting Up Oracle BI EE Shared Files"
- Section 9.2.2, "Setting the Location of the Shared BI Publisher Configuration Folder"

9.2.1 Setting Up Oracle BI EE Shared Files

This section contains the following topics:

- Section 9.2.1.1, "Specifying the RPD Publishing Directory"
- Section 9.2.1.2, "Setting the Location of the Shared Oracle BI Presentation Catalog"
- Section 9.2.1.3, "Setting the Location of the Global Cache"

9.2.1.1 Specifying the RPD Publishing Directory

Specify a repository publishing directory for the Oracle BI repository. This location is used for propagating online repository changes in a cluster.

Perform the following steps to specify the publishing directory:

1. Log in to Fusion Middleware Control.
2. Expand the **Business Intelligence** node in the Farm_*domain_name* window.
3. Click **coreapplication**.

4. Click **Deployment**, then **Repository**.
5. Click **Lock and Edit Configuration**.
6. Select **Share Repository** and specify the **RPD Publishing Directory** for the Oracle BI Repository.

   In a Windows environment, you must specify a UNC path name.
7. Click **Apply**.
8. Click **Activate Changes**.

9.2.1.2 Setting the Location of the Shared Oracle BI Presentation Catalog

Each Presentation Services instance loads the Oracle BI Presentation Catalog from the catalog location that is specified in Fusion Middleware Control.

Perform the following steps to set the location of the catalog:

1. Copy the existing (locally published) catalog to the shared location. An example of a locally published catalog is:

   ```
   ORACLE_INSTANCE/bifoundation/OracleBIPresentationServicesComponent/
   coreapplication_obipsn/catalog/SampleAppLite
   ```

   You must perform this step before designating the **Catalog Location** from Fusion Middleware Control.
2. Log in to Fusion Middleware Control.
3. Expand the **Business Intelligence** node in the Farm_*domain_name* window.
4. Click **coreapplication**.
5. Click **Deployment**, then click **Repository**.
6. Click **Lock and Edit Configuration**.
7. Specify the **Catalog Location** for the shared Oracle BI Presentation Catalog.

   In a Windows environment, you must specify a UNC path name.
8. Click **Apply**.
9. Click **Activate Changes**.

9.2.1.3 Setting the Location of the Global Cache

The global cache resides on a shared file system (a mounted file system on UNIX or a network shared drive on Windows) and stores purging events, seeding events (often generated by agents), and result sets that are associated with seeding events. Note that each Oracle BI Server still maintains its own local query cache for regular queries.

Perform the following steps to set the location of the cache:

1. Log in to Fusion Middleware Control.
2. Expand the **Business Intelligence** node in the Farm_*domain_name* window.
3. Click **coreapplication**.
4. Click **Capacity Management**, then **Performance**.
5. Click **Lock and Edit Configuration**.

6. In the Global Cache section, specify the shared location for storing purging and seeding cache entries in the **Global cache path** field. In a Windows environment, you must specify a UNC path name.
7. Enter a value for **Global cache size** to specify the maximum size of the global cache (for example, 250 MB).
8. Click **Apply**.
9. Click **Activate Changes**.
10. Click **Restart to apply recent changes**.
11. Click **Restart** under Manage System.
12. Click **Yes** in the confirmation dialog.

9.2.2 Setting the Location of the Shared BI Publisher Configuration Folder

Perform the following steps to set server configuration options for BI Publisher:

1. Copy the contents of the *`DOMAIN_HOME`*`/config/bipublisher/repository` directory to the shared configuration folder location.
2. On APPHOST1, log in to BI Publisher with Administrator credentials and select the **Administration** tab.
3. Under System Maintenance, select **Server Configuration**.
4. In the **Path** field under Configuration Folder, enter the shared location for the Configuration Folder.
5. In the **BI Publisher Repository** field under Catalog, enter the shared location for the BI Publisher Repository.
6. Apply your changes and restart the BI Publisher application, using the following steps:
   a. Log in to the Administration Console.
   b. Click **Deployments** in the Domain Structure window.
   c. Select **bipublisher(11.1.1)**.
   d. Click **Stop** and then select **When work completes** or **Force Stop Now**.
   e. After the application has stopped, click **Start** and then select **Servicing all requests**.
7. Because BI Publisher reads its configuration from the Administration Server central location rather than from the Managed Server's configuration directory when the Managed Servers are restarted, you must copy the XML configuration file for BI Publisher from the Managed Server to the Administration Server location.

   To do this, on APPHOST1, copy the xmlp-server-config.xml file from:

   ```
   ORACLE_BASE/admin/domain_name/mserver/domain_name/config/bipublisher
   ```

   to:

   ```
   ORACLE_BASE/admin/domain_name/aserver/domain_name/config/bipublisher
   ```

9.3 Scaling Out the Oracle Business Intelligence System on APPHOST2

This section explains how to scale out Oracle Business Intelligence on APPHOST2. Perform the steps in the following sections:

- Section 9.3.1, "Using the Configuration Assistant to Scale Out the Oracle BI System"
- Section 9.3.2, "Scaling Out the System Components"
- Section 9.3.3, "Configuring Secondary Instances of Singleton System Components"
- Section 9.3.4, "Using Oracle Essbase Studio Server in a Highly Availability Environment"

9.3.1 Using the Configuration Assistant to Scale Out the Oracle BI System

Perform the following steps to run the Configuration Assistant from the Oracle home directory to scale out the Oracle BI system:

1. Ensure that the Oracle BI System on APPHOST1 is up and running.
2. Change the directory to the location of the Configuration Assistant, as follows:

   ```
   APPHOST2> cd ORACLE_HOME/bin
   ```

3. Start the Oracle Business Intelligence Configuration Assistant using the following command:

   ```
   APPHOST2> ./config.sh
   ```

4. In the Welcome screen, click **Next**.
5. In the Prerequisite Checks screen, verify that all checks complete successfully, and click **Next**.
6. In the Create, Scale Out, or Extend screen, select **Scale Out BI System** and enter the following:

   - **Host Name:** `ADMINVHN`
   - **Port:** `7001`
   - **User name:** `weblogic`
   - **User Password:** *`your_password`*

   Click **Next**.

7. In the Scale Out BI System Details screen, enter the following:

   - **Middleware Home:** *`ORACLE_BASE`*`/product/fmw` (dimmed)
   - **Oracle Home:** *`ORACLE_BASE`*`/product/fmw/Oracle_BI1` (dimmed)
   - **WebLogic Server Home:** *`ORACLE_BASE`*`/product/fmw/wlserver_10.3` (dimmed)
   - **Domain Home:** *`ORACLE_BASE`*`/admin/`*`domain_name`*`/mserver/`*`domain_name`*
   - **Applications Home:** *`ORACLE_BASE`*`/admin/`*`domain_name`*`/mserver/applications`
   - **Instance Home:** *`ORACLE_BASE`*`/admin/instance2`
   - **Instance Name:** `instance2` (dimmed)

   Click **Next**.

8. In the Configure Ports screen, select one of the following:
   - Auto Port Configuration
   - Specify Ports using Configuration File

   Click **Next**.
9. In the Specify Security Updates screen, specify whether you want to receive security updates from Oracle Support and if you do, enter your e-mail address.

   Click **Next**.
10. In the Summary screen, click **Configure**.
11. In the Configuration Progress screen, verify that all the Configuration Tools have completed successfully and click **Next**.
12. In the Complete screen, click **Finish**.

Usually, Node Manager is started automatically when the config.sh process completes. If Node Manager is not running, then perform the following steps to start it on APPHOST2:

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:

```
APPHOST2> cd ORACLE_COMMON_HOME/common/bin
APPHOST2> ./setNMProps.sh
```

> **Note:** You must use the StartScriptEnabled property to avoid class loading failures.

2. Start Node Manager:

```
APPHOST2> cd WL_HOME/server/bin
APPHOST2> ./startNodeManager.sh
```

9.3.2 Scaling Out the System Components

Perform the following steps to scale out the system components:

1. Log in to Fusion Middleware Control.
2. Expand the **Business Intelligence** node in the Farm_*domain_name* window.
3. Click **coreapplication**.
4. Click **Capacity Management**, then click **Scalability**.
5. Click **Lock and Edit Configuration**.
6. For the APPHOST2 instance2 Oracle instance, increment the Oracle Business Intelligence components by 1:
   - BI Servers
   - Presentation Services
   - JavaHosts
7. Click **Apply**.
8. Click **Activate Changes**.

You do not need to restart anything at this point, because you perform a restart after completing the steps in Section 9.3.3.

9.3.3 Configuring Secondary Instances of Singleton System Components

The Oracle BI Cluster Controller, Oracle BI Scheduler, and Essbase Agent are singleton components that operate in active/passive mode. Configure a secondary instance of these components so that they are distributed for high availability.

Perform the following steps to configure secondary instances:

1. Log in to Fusion Middleware Control at http://admin.mycompany.com/em.
2. Expand the **Business Intelligence** node in the Farm_*BI_domain_name* window.
3. Click **coreapplication**.
4. Click **Availability**, then click **Failover**.
5. Click **Lock and Edit Configuration** to activate the Primary/Secondary Configuration section of the **Availability** tab.
6. Specify the **Secondary Host/Instance** for BI Scheduler, BI Cluster Controller, and Essbase Agent.
7. In the Essbase Agents section, ensure that the **Shared Folder Path** is set to `ORACLE_BASE/admin/domain_name/cluster_name/Essbase/essbaseserver1`.

   Note: You must manually copy the contents of the `ORACLE_INSTANCE/Essbase/essbaseserver1` directory to the shared folder path.
8. Click **Apply**.
9. Click **Activate Changes**.

   Under Potential Single Points of Failure, it reports **No problems - all components have a backup**.
10. Click **Restart to apply recent changes**.
11. Click **Restart** under Manage System.
12. Click **Yes** in the confirmation dialog.

9.3.4 Using Oracle Essbase Studio Server in a Highly Availability Environment

Oracle Essbase Studio Server does not support high-availability, which can cause issues during failover. For example, suppose that Essbase Studio Server is configured on APPHOST1, which then crashes. The system fails over to APPHOST2, but Essbase Studio Server is not running there. You cannot simply start the server on APPHOST2, because the Essbase Studio catalog database should not be used by two or more Essbase Studio Server instances, either simultaneously or in succession. Oracle strongly recommends that each Essbase Studio Server point to its own unique catalog database.

If you start Essbase Studio Server on a computer that is different from the computer on which you ran the cube deployment, then drill-through reports do not work as expected. To work around this issue, perform the following steps to manually update the Essbase Studio Server information from the Essbase Studio Console:

1. Go to **Tools**, then select **Update Cube Linkage**.
2. Update the information in the Cube Linkage Essbase Studio Server column to reflect the correct Essbase Studio Server.

See the documentation for Essbase Studio Server at the following location for complete information on starting and using it with a catalog database:

`http://docs.oracle.com/cd/E17236_01/nav/portal_3.htm`

9.4 Configuring the bi_server2 Managed Server

This section explains how to configure the bi_server2 Managed Server, and contains the following topics:

- Section 9.4.1, "Setting the Listen Address for the bi_server2 Managed Server"
- Section 9.4.2, "Disabling Host Name Verification for the bi_server2 Managed Server"

9.4.1 Setting the Listen Address for the bi_server2 Managed Server

Ensure that you have performed the steps that are described in Section 3.4.2, "Enabling Virtual IPs for the Managed Servers" before setting the bi_server2 listen address.

Perform the following steps to set the listen address for the Managed Server:

1. Log in to the Oracle WebLogic Server Administration Console.
2. In the Change Center, click **Lock & Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page is displayed.
5. Select **bi_server2** in the Names column of the table. The settings page for bi_server2 is displayed.
6. Set the **Listen Address** to APPHOST2VHN1.
7. Click **Save**.
8. Click **Activate Changes**.

   The changes do not take effect until the bi_server2 Managed Server is restarted.

9.4.2 Disabling Host Name Verification for the bi_server2 Managed Server

This step is required if you have not configured the appropriate certificates to authenticate the different nodes with the Administration Server, as described in Chapter 10, "Setting Up Node Manager for an Enterprise Deployment." If you have not configured the server certificates, then you see errors when managing the different Oracle WebLogic Servers. To avoid these errors, disable host name verification while configuring and validating the topology and enable it again after the EDG topology configuration is complete as described in Chapter 10, "Setting Up Node Manager for an Enterprise Deployment."

Perform the following steps to disable host name verification:

1. Log in to the Administration Console.
2. In the Change Center, click **Lock & Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page is displayed.
5. Select **bi_server2** in the Names column of the table. The settings page for the server is displayed.

6. Open the **SSL** tab.
7. Expand the **Advanced** section of the page.
8. Set host name verification to "None".
9. Click **Save**.
10. Click **Activate Changes**.
11. The change does not take effect until the bi_server2 Managed Server is restarted using the following steps (ensure that Node Manager is up and running):
    a. In the Summary of Servers screen, select the **Control** tab.
    b. Select **bi_server2** in the table and then click **Shutdown**.
    c. Select **bi_server2** in the table and then click **Start**.
12. Restart the BI System Components on APPHOST2, using the following steps:
    a. Log in to Fusion Middleware Control.
    b. Expand the **Business Intelligence** node in the Farm_*domain_name* window.
    c. Click **coreapplication**.
    d. On the Business Intelligence Overview page, click **Restart**.

9.5 Performing Additional Configuration for Oracle Business Intelligence Availability

This section describes additional high availability configuration tasks for Oracle BI Scheduler, Oracle RTD, BI Publisher, and Oracle BI Composer. It contains the following topics:

- Section 9.5.1, "Additional Configuration Tasks for Oracle BI Scheduler"
- Section 9.5.2, "Additional Configuration Tasks for Oracle RTD"
- Section 9.5.3, "Additional Configuration Tasks for BI Publisher"
- Section 9.5.4, "Additional Configuration Tasks for Oracle BI Composer"
- Section 9.5.5, "Additional Configuration Tasks for Essbase"

9.5.1 Additional Configuration Tasks for Oracle BI Scheduler

If you use server-side scripts with Oracle BI Scheduler, then it is recommended that you configure a shared directory for the scripts so that they can be shared by all Oracle BI Scheduler components in a cluster.

Perform the following steps only if you are using server-side scripts.

Perform the following steps to share Oracle BI Scheduler scripts:

1. Copy the default Oracle BI Scheduler scripts (for example, `ORACLE_INSTANCE/bifoundation/OracleBISchedulerComponent/coreapplication_obisch1/scripts/common`) and custom Oracle BI Scheduler scripts (for example, `ORACLE_INSTANCE/bifoundation/OracleBISchedulerComponent/coreapplication_obisch1/scripts/scheduler`) from APPHOST1 to the shared BI Scheduler scripts location.

2. Update the SchedulerScriptPath and DefaultScriptPath elements of the Oracle BI Scheduler instanceconfig.xml file, as follows:

   - SchedulerScriptPath: Refers to the path where Oracle BI Scheduler-created job scripts are stored. Change this to the path of the shared BI Scheduler scripts location.
   - DefaultScriptPath: Specifies the path where user-created job scripts (not agents) are stored. Change this to the path of the shared BI Scheduler scripts location.

   In a Windows environment, you must specify a UNC path name.

3. Restart the Oracle BI Scheduler component, as follows:

   ```
   opmnctl stopproc ias-component=coreapplication_obisch1
   opmnctl startproc ias-component=coreapplication_obisch1
   ```

The instanceconfig.xml file for Oracle BI Scheduler is located in `ORACLE_INSTANCE/config/OracleBISchedulerComponent/coreapplication_obischn`. You must update this file for each Oracle BI Scheduler component in the deployment.

9.5.2 Additional Configuration Tasks for Oracle RTD

This section contains the following topics:

- Section 9.5.2.1, "Configuring Oracle RTD Clustering Properties"
- Section 9.5.2.2, "Adding System Properties to the Server Start Tab"

9.5.2.1 Configuring Oracle RTD Clustering Properties

Perform the following steps in Fusion Middleware Control to specify cluster-specific configuration properties for Oracle RTD.

You must perform these steps only for the first node in the deployment. You do not need to set cluster-specific configuration properties for Oracle RTD for subsequent nodes.

1. Log in to Fusion Middleware Control.
2. Expand the **Application Deployments** node in the Farm_*domain_name* window.
3. Click **OracleRTD(11.1.1)(bi_cluster)**.
4. Click any node under it. For example, **OracleRTD(11.1.1)(bi_server1)**.
5. In the right pane, click **Application Deployment**, and then select **System MBean Browser**.
6. In the System MBean Browser pane, expand **Application Defined MBeans**.
7. For any one of the servers under OracleRTD, navigate to the **SDClusterPropertyManager -> Misc** MBean and set the **DecisionServiceAddress** attribute to `http://biinternal.mycompany.com`. Other servers automatically get updated with the value that you set.
8. Click **Apply**.

9.5.2.2 Adding System Properties to the Server Start Tab

After scaling out Oracle RTD, perform the following steps to add three system properties to the **Server Start** tab of each Managed Server:

1. Log in to the Administration Console.

2. In the Change Center, click **Lock & Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page is displayed.
5. Select **bi_server<1,2>** in the table. The settings page for the server is displayed.
6. Click the **Server Start** tab.
7. Add the following properties in the **Arguments** box:

   ```
   -Drtd.clusterRegistryJobIntervalMs=12000
   -Drtd.clusterDepartureThresholdMs=50000
   -Drtd.clusterDepartureThreshold2Ms=50000
   ```

8. Click **Save**.
9. Click **Activate Changes**.
10. The change does not take effect until the bi_server<1,2> Managed Servers are restarted (ensure that Node Manager is up and running):
    a. In the Summary of Servers screen, select the **Control** tab.
    b. Select **bi_server<1,2>** in the table and then click **Shutdown**.
    c. Restart **bi_server<1,2>**.
11. Restart the BI System Components:

    ```
    cd /u01/app/oracle/admin/instancen/bin
    ./opmnctl stopall
    ./opmnctl startall
    ```

Performing this task enables an instance of Oracle RTD to be migrated successfully from one host to another in the event of a failure of a Managed Server.

Even after these changes, if the server migration finishes in less than 50 seconds, then the Oracle RTD batch framework remains in an inconsistent state.

If the enterprise has deployed any RTD Inline Services that host Batch Job implementations, and if after a server migration the batch console command, "batch-names", or its brief name, "bn", shows no registered batch jobs, then you must stop and restart the Oracle RTD Batch Manager service. Perform the following steps to stop and restart the service:

1. In Fusion Middleware Control, expand the **WebLogic Domain** node in the left pane. Then, right-click **bifoundation_domain** and select **System MBean Browser**.
2. Locate the **SDPropertyManager > Misc** MBean, under **Application Defined MBeans > OracleRTD > Server:bi_server*n***.

   Ensure that you select the **Misc** MBean that corresponds to the local node where you are making the change. For example, if you are connecting to APPHOST1, then ensure that you update the attribute associated with bi_server1.
3. Set the BatchManagerEnabled attribute to **false** and click **Apply**.
4. Set the BatchManagerEnabled attribute back to **true** and click **Apply**. Performing this task causes the Batch Manager to stop and be restarted.

   When the Batch Manager restarts, it runs on either the same server as before, or on a different server.
5. After restarting Batch Manager, note that the corresponding MBean does not always immediately get refreshed on the server where Batch Manager is restarted,

so this is not a concern. Instead, verify that Batch Manager is now operational by using the Batch Console tool, as in the follow steps:

a. Locate the zip file for the Oracle RTD client tools in the following location:

```
ORACLE_HOME/clients/rtd/rtd_client_11.1.1.zip
```

b. Because most Oracle RTD client tools do not run on UNIX, unzip this file in a location on a Windows computer (referred to here as *RTD_HOME*). Then, locate the batch console jar file in:

```
RTD_HOME/client/Batch/batch-console.jar
```

c. Change to this directory and execute the jar, passing to it the URL and port of either the Managed Server, or of the cluster proxy:

```
java -jar batch-console.jar -url http://SERVER:PORT
```

d. When prompted, enter the user name and password of a user who is a member of the Administrator role, BI_Adminstrator role, or some other role authorized to administer Oracle RTD batch jobs.

e. When prompted for a command, enter `bn`:

```
Checking server connection...
command: bn
    CrossSellSelectOffers
command:quit
```

If Batch Manager has successfully restarted, then the "bn" command lists the names of all batch implementations hosted by all deployed RTD Inline Services.

The commonly deployed example, CrossSell, hosts a batch implementation named CrossSellSelectOffers, shown in the preceding example.

9.5.3 Additional Configuration Tasks for BI Publisher

This section contains the following topics:

- Section 9.5.3.1, "Updating the BI Publisher Scheduler Configuration"
- Section 9.5.3.2, "Configuring Integration with Oracle BI Presentation Services"
- Section 9.5.3.3, "Setting the Oracle BI EE Data Source"

9.5.3.1 Updating the BI Publisher Scheduler Configuration

Perform the following tasks to update the configuration for BI Publisher Scheduler:

- Section 9.5.3.1.1, "Updating the Quartz Configuration"
- Section 9.5.3.1.2, "Updating the WebLogic JNDI URL for BI Publisher Scheduler"
- Section 9.5.3.1.3, "Configuring JMS for BI Publisher"
- Section 9.5.3.1.4, "Updating the JMS Shared Temp Directory"

9.5.3.1.1 Updating the Quartz Configuration Perform the following steps to set Scheduler configuration options:

1. On APPHOST1, log in to BI Publisher with Administrator credentials and select the **Administration** tab.
2. Under System Maintenance, select **Scheduler Configuration**.

3. Select **Quartz Clustering** under the Scheduler Selection.
4. Click **Apply**.

9.5.3.1.2 Updating the WebLogic JNDI URL for BI Publisher Scheduler

Perform the following steps to update the WebLogic JNDI URL for the BI Publisher Scheduler:

1. Log in to BI Publisher at the following URL:

   http://APPHOST1VHN1:9704/xmlpserver

2. Click the **Administration** link.
3. Click **Scheduler Configuration** under System Maintenance. The Scheduler Configuration screen is displayed.
4. Update the **WebLogic JNDI URL** under JMS Configuration, as follows:

   ```
   cluster:t3://bi_cluster
   ```

5. Click **Test JMS**.

   You see a confirmation message that JMS tested successfully.

6. Click **Apply**. The changes are sent to the cluster to be applied at runtime.
7. Check the Scheduler status from the **Scheduler Diagnostics** tab.

9.5.3.1.3 Configuring JMS for BI Publisher

You must configure the location for all persistence stores to a directory that is visible from both nodes. Change all persistent stores to use this shared base directory. Perform the following steps to configure JMS:

1. Log into the Administration Console.
2. In the Domain Structure window, expand the **Services** node and click the **Persistent Stores** node. The Summary of Persistent Stores page is displayed.
3. In the Change Center, click **Lock & Edit**.
4. Click **New** and **Create File Store**.
5. Enter a name (for example, BipJmsStore2) and target BI_SERVER2. Enter a directory that is located in shared storage so that it is accessible from both APPHOST1 and APPHOST2:

   ```
   ORACLE_BASE/admin/domain_name/bi_cluster/jms
   ```

6. Click **OK** and **Activate Changes**.
7. In the Domain Structure window, expand the **Services** node and click the **Messaging > JMS Servers** node. The Summary of JMS Servers page is displayed.
8. In the Change Center, click **Lock & Edit**.
9. Click **New**.
10. Enter a name (for example, BipJmsServer2) and in the **Persistence Store** drop-down list, select **BipJmsStore2** and click **Next**.
11. Select **BI_SERVER2** as the target.
12. Click **Finish** and **Activate Changes**.
13. In the Domain Structure window, expand the **Services** node and click the **Messaging > JMS Modules** node. The JMS Modules page is displayed.
14. In the Change Center, click **Lock & Edit**.

15. Click **BipJmsResource** and click the **Subdeployments** tab.
16. Select **BipJmsSubDeployment** under Subdeployments.
17. Add the new BI Publisher JMS Server, **BipJmsServer2**, as an additional target for the subdeployment.
18. Click **Save** and **Activate Changes**.

To validate the JMS configuration performed for BI Publisher, perform the steps in Section 9.5.3.1.4, "Updating the JMS Shared Temp Directory."

9.5.3.1.4 Updating the JMS Shared Temp Directory Follow the steps in this section to update the JMS Shared Temp Directory for the BI Publisher Scheduler. You need to perform the steps in this section on only one of the APPHOSTS (either APPHOST1 or APPHOST2).

Perform the following steps to update the BI Publisher Scheduler configuration:

1. Log in to BI Publisher at the one of the following URLs:
   - http://APPHOST1VHN1:9704/xmlpserver
   - http://APPHOST2VHN1:9704/xmlpserver
2. Click the **Administration** link.
3. Click **Scheduler Configuration** under System Maintenance. The Scheduler Configuration screen is displayed.
4. Update the **Shared Directory** by entering a directory that is located in the shared storage. This shared storage is accessible from both APPHOST1 and APPHOST2.
5. Click **Test JMS**.

   You see a confirmation message that JMS tested successfully.

   > **Note:** If you do not see a confirmation message for a successful test, then verify that the JNDI URL is set to the following:
   >
   > ```
   > cluster:t3://bi_cluster
   > ```

6. Click **Apply**.
7. Check the Scheduler status from the **Scheduler Diagnostics** tab.

9.5.3.2 Configuring Integration with Oracle BI Presentation Services

Perform the following steps to configure BI Publisher integration with Oracle BI Presentation Services:

1. Log into BI Publisher with Administrator credentials and select the **Administration** tab.
2. Under **Integration**, select **Oracle BI Presentation Services**.
3. Verify and update the following:
   - **Server Protocol:** HTTP
   - **Server:** biinternal.mycompany.com
   - **Port:** 80
   - **URL Suffix:** analytics-ws/saw.dll

4. Click **Apply**.
5. Restart the BI Publisher application.

9.5.3.3 Setting the Oracle BI EE Data Source

The Oracle BI EE Data Source must point to the clustered Oracle BI Servers through the Cluster Controllers. Perform this task in BI Publisher.

Perform the following steps to set the Oracle BI EE data source in BI Publisher:

1. Log in to BI Publisher with Administrator credentials and select the **Administration** tab.
2. Under **Data Sources**, select **JDBC Connection**.
3. Update the Oracle BI EE data source setting by changing the **Connection String** parameter to the following:

   ```
   jdbc:oraclebi://primary_cluster_controller_host:primary_cluster_controller_
   port/PrimaryCCS=primary_cluster_controller_host;PrimaryCCSPort=primary_cluster_
   controller_port;SecondaryCCS=secondary_cluster_controller_host;
   SecondaryCCSPort=secondary_cluster_controller_port;
   ```

   For example:

   ```
   jdbc:oraclebi://APPHOST1:9706/PrimaryCCS=APPHOST1;PrimaryCCSPort=9706;
   SecondaryCCS=APPHOST2;SecondaryCCSPort=9706;
   ```

4. Select **Use System User**.

   If you do not want to use the system user for the connection, then deselect **Use System User** and specify the BIImpersonateUser credentials for **Username** and **Password**.

   For more information about the BIImpersonateUser user in this context, see "Credentials for Connecting to the Oracle BI Presentation Catalog" in *Oracle Fusion Middleware Developer's Guide for Oracle Business Intelligence Enterprise Edition*.
5. Click **Test Connection**. You see a "Connection established successfully" message.
6. Click **Apply**.

9.5.4 Additional Configuration Tasks for Oracle BI Composer

Perform the following steps for Oracle BI Composer to change the port value to the load balancer port:

1. Log in to Fusion Middleware Control Console.
2. Expand Application Deployments in the left-hand navigation pane.
3. Under bicomposer(11.1.1) (bi_cluster), right-click **bicomposer(11.1.1) (bi_server1)** and select **System MBean Browser**.
4. Go to the following MBean:

   Application Defined MBeans > oracle.adf.share.connections > Server: bi_server1 > Application: bicomposer > ADFConnections > BISoapConnection > bi-default
5. Set the **Protocol** attribute to **http**.
6. Set the **Port** attribute to the load balancer HTTP port (80).
7. Set the **Host** attribute to the internal load balancer URL as follows:

```
biinternal.mycompany.com
```

8. Click **Apply**.
9. Go to the **ADFConnections** MBean and select the Operations tab.
10. Click **Save** and click **Invoke** on the resulting screen.
11. Restart the BI Composer application using Fusion Middleware Control or the Administration Console.

9.5.5 Additional Configuration Tasks for Essbase

Perform the following steps for Essbase to change the Merant (Data Direct) driver version to 7.*x* when the repository database is Oracle 12*c*.

On APPHOST1 and APPHOST2:

1. Run the Data Direct script to list the installed Merant (Data Direct) driver versions:

```
$ORACLE_HOME/bin/datadirect_version.sh -l
5.3
6.0
6.1
7.0
7.0.1
7.1.2
```

2. Set the current Merant (Data Direct) driver version to 7.*x*.

```
$ORACLE_HOME/bin/datadirect_version.sh -i <Oracle BI Instance
Directory> -s <Driver Version>.
```

3. Restart the system components.

```
$ cd /u01/app/oracle/admin/instancen/bin
$ ./opmnctl stopall
$ ./opmnctl startall
```

9.6 Other Post-Configuration and Verification Tasks

After performing the high availability configuration tasks for Oracle BI Scheduler, Oracle RTD, BI Publisher, and Oracle BI Composer, follow these additional instructions for post-configuration and verification.

This section contains the following topics:

- Section 9.6.1, "Configuring a Default Persistence Store for Transaction Recovery"
- Section 9.6.2, "Starting and Validating Oracle Business Intelligence on APPHOST2"
- Section 9.6.3, "Configuring Node Manager for the Managed Servers"
- Section 9.6.4, "Configuring Server Migration for the Managed Servers"

9.6.1 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 might 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 that is accessible to a server and its backup servers.

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

Perform the following steps for each Managed Server to set the location for the default persistence store:

1. Log in to the Administration Console.
2. In the Change Center, click **Lock & Edit**.
3. In the Domain Structure window, expand the **Environment** node and click the **Servers** node. The Summary of Servers page is displayed.
4. Click the name of the server (represented as a hyperlink) in the Names column of the table. The Settings page for the selected server is displayed, and defaults to the **Configuration** tab.
5. Open the **Services** tab.
6. In the Default Store section of the page, enter the path to the folder where the default persistent stores store its data files. The directory structure of the path is as follows:

   ```
   ORACLE_BASE/admin/domain_name/bi_cluster_name/tlogs
   ```

   Use the same path for each Managed Server. When the Managed Servers are restarted, subdirectories are created for each one.
7. Click **Save** and **Activate Changes**.
8. Restart both Managed Servers.

Note: To enable migration of the Transaction Recovery service, specify a location on a persistent storage solution that is available to other servers in the cluster. Both bi_server1 and bi_server2 must be able to access this directory.

9.6.2 Starting and Validating Oracle Business Intelligence on APPHOST2

This section contains the following topics:

- Section 9.6.2.1, "Starting the bi_server2 Managed Server"
- Section 9.6.2.2, "Starting the Oracle Business Intelligence System Components"
- Section 9.6.2.3, "Validating Oracle Business Intelligence URLs"
- Section 9.6.2.4, "Validating Access Through the Load Balancer"
- Section 9.6.2.5, "Validating Essbase Clustering"

9.6.2.1 Starting the bi_server2 Managed Server

Perform the following steps to start the bi_server2 Managed Server:

1. Start the bi_server2 Managed Server using the Administration Console, as in the following steps:

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

   b. Click **Servers**. The Summary of Servers page is displayed.

   c. Click the **Control** tab.

d. Select **bi_server2** and then click **Start**.

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

9.6.2.2 Starting the Oracle Business Intelligence System Components

You can control Oracle Business Intelligence system components using opmnctl commands.

Perform the following steps to start the Oracle Business Intelligence system components using the opmnctl command-line tool:

1. Go to the directory that contains the OPMN command-line tool, located in `ORACLE_INSTANCE/bin`.
2. Run the opmnctl command to start the Oracle Business Intelligence system components:
   - `opmnctl startall`

     Starts OPMN and all Oracle Business Intelligence system components.
   - `opmnctl start`

     Starts OPMN only.
   - `opmnctl startproc ias-component=component_name`

     Starts a particular system component. For example, where coreapplication_obips2 is the Presentation Services component:

     ```
     opmnctl startproc ias-component=coreapplication_obips2
     ```
3. Check the status of the Oracle Business Intelligence system components:

   ```
   opmnctl status
   ```

9.6.2.3 Validating Oracle Business Intelligence URLs

Access the following URLs:

- Access http://APPHOST2VHN1:9704/analytics to verify the status of bi_server1.
- Access http://APPHOST2VHN1:9704/wsm-pm to verify the status of Web Services Manager. Click **Validate Policy Manager**. A list of policies and assertion templates available in the data is displayed.

  Note: The configuration is incorrect if no policies or assertion templates are displayed.
- Access http://APPHOST2VHN1:9704/xmlpserver to verify the status of the BI Publisher application.
- Access http://APPHOST2VHN1:9704/ui to verify the status of the Oracle RTD application.
- Access http://APPHOST2VHN1:9704/mapviewer to verify the status of Oracle MapViewer.
- Access http://APPHOST2VHN1:9704/aps/Essbase to verify the status of the Oracle Essbase application.
- Access http://APPHOST2VHN1:9704/aps/SmartView to verify the status of the Smart View application.

- Access http://APPHOST2VHN1:9704/workspace to verify the status of the Workspace application.
- Access http://APPHOST2VHN1:9704/hr to verify the status of the Financial Reporting application.
- Access http://APPHOST2VHN1:9704/calcmgr/index.htm to verify the status of the Calculation Manager application.

9.6.2.4 Validating Access Through the Load Balancer

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

1. While bi_server2 is running, stop bi_server1 using the Administration Console.
2. Access the following URLs to verify that routing and failover is functioning properly:
   - http://bi.mycompany.com/analytics
   - http://bi.mycompany.com/xmlpserver
   - http://bi.mycompany.com/ui
3. Start bi_server1 from the Administration Console.
4. Stop bi_server2 from the Administration Console.
5. Access the following URLs to verify that routing and failover is functioning properly:
   - http://bi.mycompany.com/analytics
   - http://bi.mycompany.com/xmlpserver
   - http://bi.mycompany.com/ui
6. Start bi_server2 from the Administration Console.

9.6.2.5 Validating Essbase Clustering

Perform the following steps to validate Essbase clustering:

1. Check the APS (Hyperion Provider Services) test URL:

   https://bi.mycompany.com/aps/Essbase?ClusterName=EssbaseCluster-1
2. Run the following command on APPHOST1:

   ```
   ORACLE_BASE/admin/instance1/bin/opmnctl stopproc
   ias-component=essbaseserver1
   ```
3. Ensure that Essbase starts on APPHOST2:

   ```
   ORACLE_BASE/admin/instance2/bin/opmnctl status
   ```

   The status is `init`, then `Alive`.
4. Check the APS test URL again:

   https://bi.mycompany.com/aps/Essbase?ClusterName=EssbaseCluster-1

9.6.3 Configuring Node Manager for the Managed Servers

Oracle recommends using host name verification for the communication between Node Manager and the servers in the domain. This verification requires the use of certificates for the different addresses that communicate with the Administration

Server and other servers. See Chapter 10, "Setting Up Node Manager for an Enterprise Deployment" for further details. The procedures in that chapter must be performed twice using the information that is provided in Table 9–2.

Table 9–2 Details for Host Name Verification for Node Manager and Servers

Run	Host Name (HOST)	Server Name (WLS_SERVER)
Run1:	APPHOST1	bi_server1
Run2:	APPHOST2	bi_server2

9.6.4 Configuring Server Migration for the Managed Servers

Server Migration is required for proper failover of the BI Publisher components in the event of failure in any of the APPHOST1 and APPHOST2 nodes. See Chapter 11, "Configuring Server Migration for an Enterprise Deployment" for further details.

9.7 Backing Up the Installation

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

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

1. Back up the web tier, using the following steps:

   a. Shut down the instance using `opmnctl`:

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

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

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

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

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

   Repeat this step for WEBHOST2.

   d. Start the instance using `opmnctl`:

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

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

3. Back up the BI Instance in the application tier, using the following steps:

   a. Shut down the instance using opmnctl:

      ```
      APPHOSTn> ORACLE_INSTANCE/bin/opmnctl stopall
      ```

   b. Back up the Middleware home on the application tier using the following command:

      ```
      APPHOSTn> tar -cvpf BACKUP_LOCATION/bi.tar MW_HOME
      ```

   c. Back up the Oracle instance on the application tier using the following command:

      ```
      APPHOSTn> tar -cvpf BACKUP_LOCATION/bi_instance_name.tar ORACLE_INSTANCE
      ```

   d. Start the instance using opmnctl:

      ```
      APPHOSTn> ORACLE_INSTANCE/bin/opmnctl startall
      ```

4. Back up the Administration Server and Managed Server domain directories to save the domain configuration. The configuration files all exist in the *ORACLE_BASE*/admin/*domain_name* directory. Run the following command to create the backup:

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

   Note: Create backups on all computers in the application tier by following the steps that are described in this section.

10 Setting Up Node Manager for an Enterprise Deployment

This chapter describes how to configure Node Manager according to the Enterprise Deployment recommendations.

Important: Oracle strongly recommends that you read the *Oracle Fusion Middleware Release Notes* for any additional installation and deployment considerations before starting the setup process.

This chapter contains the following topics:

- Section 10.1, "About Setting Up Node Manager"
- Section 10.2, "Changing the Location of the Node Manager Log File"
- Section 10.3, "Enabling Host Name Verification Certificates for Node Manager"
- Section 10.4, "Starting Node Manager"

10.1 About Setting Up Node Manager

Node Manager enables you to start and stop the Administration Server and the Managed Servers.

Oracle provides two main recommendations for Node Manager configuration in enterprise deployment topologies:

1. Oracle recommends placing the Node Manager log file in a location that is different from the default one (which is inside the Middleware home where Node Manager resides). See Section 10.2, "Changing the Location of the Node Manager Log File" for further details.
2. Oracle also recommends using host name verification for the communication between Node Manager and the servers in the domain. This verification requires the use of certificates for the different addresses that are used in the domain. This chapter explains the steps for configuring certificates in the hosts for host name verification. See Section 10.3, "Enabling Host Name Verification Certificates for Node Manager" for further details.

Note: The passwords that are used in this guide are provided only as examples. Use secure passwords in a production environment. For example, use passwords that consist of random sequences of both uppercase and lowercase characters as well as numbers.

10.2 Changing the Location of the Node Manager Log File

Change the location of the Node Manager log file on all nodes of the BI domain. To change the location, edit the nodemanager.properties file, which is located in the `*MW_HOME*/wlserver_10.3/common/nodemanager` directory. Add the new location for the log file using the following line:

```
LogFile=ORACLE_BASE/admin/nodemanager.log
```

Oracle recommends that this location be outside the *MW_HOME* directory and inside the admin directory for the EDG.

Restart Node Manager for the change to take effect.

10.3 Enabling Host Name Verification Certificates for Node Manager

Configuring host name verification certificates for communication between Node Manager and the Administration Server consists of the following steps:

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

10.3.1 Generating Self-Signed Certificates Using the utils.CertGen Utility

The certificates that are added in this chapter (as an example) address a configuration in which Node Manager listens on a physical host name (APPHOST*n*.mycompany.com) and a Managed Server listens on a virtual host name (APPHOST*n*VHN1.mycompany.com). Whenever a Managed Server is using a virtual host name, it is implied that the Managed Server can be migrated from one node to another. Consequently, the directory where keystores and trust keystores are maintained ideally must reside on a shared storage that is accessible from the failover. If additional host names are used in the same or different nodes, then the steps in this example must be extended to:

- Add the required host names to the certificate stores (if they are different from APPHOST*n*.mycompany.com and APPHOST*n*VHN1.mycompany.com).
- Change the identity and trust store location information for Node Manager (if the additional host names are used by Node Manager) or for the servers (if the additional host names are used by Managed Servers).

Follow these steps to create self-signed certificates on APPHOST*n*. These certificates are created using the network name or alias. For information on using trust CA certificates instead, see "Configuring Identity and Trust" in *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

The following examples configure certificates for APPHOST*n*.mycompany.com and APPHOST*n*VHN1.mycompany.com; that is, it is assumed that both a physical host name (APPHOST*n*) and a virtual host name (APPHOST*n*VHN1) are used in APPHOST*n*. It is also assumed that APPHOST*n*.mycompany.com is the address that is used by Node Manager, and APPHOST*n*VHN1.mycompany.com is the address that is used by a Managed Server or the Administration Server. This is the common situation for nodes that host an Administration Server and an Oracle Fusion Middleware component, or for nodes on which two Managed Servers coexist with one server listening on the physical host name and one server using a virtual host name (which is the case for servers that use migration servers).

1. Configure the environment by running the *WL_HOME*/server/bin/setWLSEnv.sh script. In the Bourne shell, run the following commands:

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

   Verify that the CLASSPATH environment variable is set: by using the following command:

   ```
   APPHOSTn> echo $CLASSPATH
   ```

2. Create a user-defined directory for the certificates. For example, create a directory called `certs` under the *ORACLE_BASE*/admin/*domain_name*/*cluster_name* directory. Note that certificates can be shared across WLS domains.

   ```
   APPHOSTn> cd ORACLE_BASE/admin/domain_name/cluster_name
   APPHOSTn> mkdir certs
   ```

   > **Note:** 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 that are used for different purposes (such as SSL configured for HTTP invocations).

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

   ```
   APPHOSTn> cd certs
   ```

4. Run the utils.CertGen tool from the user-defined directory to create the certificates for both APPHOST*n*.mycompany.com and APPHOST*n*VHN1.mycompany.com.

   Syntax (all on a single line):

   ```
   java utils.CertGen Key_Passphrase Cert_File_Name Key_File_Name
   [export | domestic] [Host_Name]
   ```

   Examples:

   ```
   APPHOSTn> java utils.CertGen password APPHOSTn.mycompany.com_cert
   APPHOSTn.mycompany.com_key domestic APPHOSTn.mycompany.com

   APPHOSTn> java utils.CertGen password APPHOSTnVHN1.mycompany.com_cert
   APPHOSTnVHN1.mycompany.com_key domestic APPHOSTnVHN1.mycompany.com
   ```

   Sample output for the command that is shown in the first example is:

   ```
   ...... Will generate certificate signed by CA from CertGenCA.der file
   ...... With Domestic Key Strength
   ...... Common Name will have Hostname APPHOSTn.mycompany.com
   ...... Issuer CA name is CN=CertGenCAB,OU=FOR TESTING ONLY,O=MyOrganization,
   L=MyTown,ST=MyState,C=US
   ```

10.3.2 Creating an Identity Keystore Using the utils.ImportPrivateKey Utility

Follow these steps to create an identity keystore on APPHOST*n*:

1. Create a new identity keystore called appIdentityKeyStore using the utils.ImportPrivateKey utility. Create this keystore under the same directory as the certificates (that is, `ORACLE_BASE/admin/domain_name/cluster_name/ certs`).

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

2. Import the certificate and private key for both APPHOST*n*.mycompany.com and APPHOST*n*VHN1.mycompany.com into the identity store. Ensure that you use a different alias for each of the certificate/key pairs imported.

   Syntax (all on a single line):

   ```
   java utils.ImportPrivateKey Keystore_File Keystore_Password
   Certificate_Alias_to_Use Private_Key_Passphrase
   Certificate_File Private_Key_File [Keystore_Type]
   ```

   Examples:

   ```
   APPHOSTn> java utils.ImportPrivateKey appIdentityKeyStore.jks password
   appIdentity1 password ORACLE_BASE/admin/domain_name/cluster_name/
   certs/APPHOSTn.mycompany.com_cert.pem ORACLE_BASE/admin/domain_name/
   cluster_name/certs/APPHOSTn.mycompany.com_key.pem

   APPHOSTn> java utils.ImportPrivateKey appIdentityKeyStore.jks password
   appIdentity1 password ORACLE_BASE/admin/domain_name/
   cluster_name/certs/APPHOSTnVHN1.mycompany.com_cert.pem ORACLE_BASE/admin/
   domain_name/cluster_name/certs/APPHOSTnVHN1.mycompany.com_key.pem
   ```

10.3.3 Creating a Trust Keystore Using the Keytool Utility

You need to perform the steps in this section only for the first Managed Server.

Perform the following steps to create the trust keystore on APPHOST1:

1. Copy the standard Java keystore to create the new trust keystore because it already contains most of the root CA certificates that are needed. Oracle does not recommend modifying the standard Java trust keystore directly. Copy the standard Java keystore CA certificates that are located in the `WL_HOME/server/lib` directory to the same directory as the certificates. For example:

   ```
   APPHOST1> cp WL_HOME/server/lib/cacerts ORACLE_BASE/admin/domain_name/cluster_
   name/certs/appTrustKeyStore.jks
   ```

2. The default password for the standard Java keystore is "changeit". Oracle recommends always changing the default password. Use the keytool utility to do this. The syntax is (all on a single line):

   ```
   APPHOST1> keytool -storepasswd -new New_Password -keystore Trust_Keystore
   -storepass Original_Password
   ```

   For example:

   ```
   APPHOST1> keytool -storepasswd -new password -keystore appTrustKeyStore.jks
   -storepass changeit
   ```

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

   ```
   APPHOST1> keytool -import -v -noprompt -trustcacerts -alias Alias_Name
   -file CA_File_Location -keystore Keystore_Location -storepass Keystore_Password
   ```

   For example:

   ```
   APPHOST1> keytool -import -v -noprompt -trustcacerts -alias clientCACert -file
   WL_HOME/server/lib/CertGenCA.der -keystore appTrustKeyStore.jks -storepass
   password
   ```

10.3.4 Configuring Node Manager to Use the Custom Keystores

To configure Node Manager to use the custom keystores, add the following lines to the end of the nodemanager.properties file located in the *WL_HOME*/common/nodemanager directory:

```
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityKeyStoreFileName=Identity_Keystore
CustomIdentityKeyStorePassPhrase=Identity_Keystore_Password
CustomIdentityAlias=Identity_Keystore_Alias
CustomIdentityPrivateKeyPassPhrase=Private_Key_Used_When_Creating_Certificate
```

Ensure that you use the correct value for CustomIdentityAlias on each node. For example, on APPHOST1, use appIdentity1.

For example, on APPHOST1:

```
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityKeyStoreFileName=ORACLE_BASE/admin/domain_name/cluster_name/
certs/appIdentityKeyStore.jks
CustomIdentityKeyStorePassPhrase=password
CustomIdentityAlias=appIdentity1
CustomIdentityPrivateKeyPassPhrase=password
```

The passphrase entries in the nodemanager.properties file get encrypted when you start Node Manager as described in Section 10.4, "Starting Node Manager." For security reasons, you want to minimize the time that 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 get encrypted.

10.3.5 Configuring Managed Servers to Use the Custom Keystores

You must perform the steps in this section for the Administration Server and all Managed Servers.

Perform the following steps to configure the identity and trust keystores:

1. Log in to the Oracle WebLogic Server Administration Console.
2. In the Change Center, click **Lock & Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page is displayed.
5. Click the name of the server for which you want to configure the identity and trust keystores (bi_server*n*). The settings page for the selected server is displayed.
6. Select **Configuration** and **Keystores**.

7. In the **Keystores** field, change to the **Custom Identity and Custom Trust** method for storing and managing private keys/digital certificate pairs and trusted CA certificates.

8. In the Identity section, define attributes for the identity keystore using the following steps:

   a. **Custom Identity Keystore:** Enter the fully qualified path to the identity keystore:

   ```
   ORACLE_BASE/admin/domain_name/aserver/cluster_name/certs/
   appIdentityKeyStore.jks
   ```

   b. **Custom Identity Keystore Type:** Ensure that this field is blank. It has a default value of JKS.

   c. **Custom Identity Keystore Passphrase:** Enter the keystore password (*Keystore_Password*) you provided in Section 10.3.2, "Creating an Identity Keystore Using the utils.ImportPrivateKey Utility."

   This attribute might be optional or required, depending on the type of keystore. All keystores require the passphrase to write to the keystore. However, some keystores do not require the passphrase to read from the keystore. Oracle WebLogic Server reads only from the keystore, so whether you need to define this property depends on the requirements of the keystore.

9. In the Trust section, define properties for the trust keystore using the following steps:

   a. **Custom Trust Keystore:** Enter the fully qualified path to the trust keystore:

   ```
   ORACLE_BASE/admin/domain_name/cluster_name/certs/
   appTrustKeyStore.jks
   ```

   b. **Custom Trust Keystore Type:** Ensure that this field is blank. It has a default value of **JKS**.

   c. **Custom Trust Keystore Passphrase:** Enter the password that you provided for *New_Password* in Section 10.3.3, "Creating a Trust Keystore Using the Keytool Utility."

   This attribute might be optional or required, depending on the type of keystore. All keystores require the passphrase to write to the keystore. However, some keystores do not require the passphrase to read from the keystore. Oracle WebLogic Server reads only from the keystore, so whether you define this property depends on the requirements of the keystore.

10. Click **Save**.

11. In the Change Center, click **Activate Changes**.

12. Select **Configuration** and **SSL**.

13. In the Change Center, click **Lock & Edit**.

14. In the **Private Key Alias** field, enter the alias that you used for the host name on which the Managed Server listens.

    In the **Private Key Passphrase** and the **Confirm Private Key Passphrase** fields, enter the password for the keystore that you created in Section 10.3.2, "Creating an Identity Keystore Using the utils.ImportPrivateKey Utility."

15. Click **Save**.

16. In the Change Center, click **Activate Changes**.

10.3.6 Changing the Host Name Verification Setting for the Managed Servers

After you perform the steps in the previous sections, you set host name verification for the affected Managed Servers to **Bea Host Name Verifier**. Perform the following steps to change the host name verification setting for all Managed Servers:

1. Log in to the Administration Console.
2. In the Change Center, click **Lock & Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page is displayed.
5. Select the Managed Server in the **Names** column of the table. The settings page for the server is displayed.
6. Open the **SSL** tab.
7. Expand the **Advanced** section of the page.
8. Set **Hostname Verification** to **BEA Hostname Verifier**.
9. Click **Save**.
10. In the Change Center, click **Activate Changes**.
11. Restart the Managed Server for which the changes have been applied.

10.4 Starting Node Manager

When using a common/shared storage installation for *MW_HOME*, Node Manager is started from different nodes using the same base configuration (the nodemanager.properties file). In that case, you must add the certificate for all the nodes that share the binaries to the appIdentityKeyStore.jks identity store. To do this, create the certificate for the new node and import it to appIdentityKeyStore.jks as described in Section 10.3.2, "Creating an Identity Keystore Using the utils.ImportPrivateKey Utility." After 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 do this, set different environment variables before starting Node Manager in the different nodes:

```
APPHOSTn> cd WL_HOME/server/bin
APPHOSTn> export JAVA_OPTIONS=-DCustomIdentityAlias=appIdentityn
```

Ensure that you specify the custom identity alias that is specifically assigned to each host. For example, specify appIdentity1 for APPHOST1 and appIdentity2 for APPHOST2.

Note: Verify that Node Manager is using the appropriate stores and alias from the Node Manager output. Node Manager output is displayed as follows:

```
CustomIdentityKeyStoreFileName=ORACLE_BASE/admin/domain_
name/cluster_name/certs/appIdentityKeyStore.jks
CustomIdentityAlias=appIdentityn
```

where *n* is 1, 2, ...

If you are not using a common/shared storage installation for *MW_HOME*, then perform the following steps to run commands to restart Node Manager on APPHOST*n*:

1. Stop the Node Manager process by pressing CTRL-C in the shell in which it was started, or by process identification and kill in the operating system.
2. Start Node Manager, as follows:

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

> **Note:** If you have not configured and started Node Manager for the first time, then run the `ORACLE_COMMON_HOME/common/bin/setNMProps.sh` script. This script enables the use of the start script, which is required for Oracle Business Intelligence.

11

Configuring Server Migration for an Enterprise Deployment

This chapter describes the procedures for configuring server migration for the Oracle Business Intelligence enterprise deployment.

> **Important: Oracle strongly recommends that you read the *Oracle Fusion Middleware Release Notes* for any additional installation and deployment considerations before starting the setup process.**

This chapter contains the following topics:

11.1 Overview of Server Migration for an Enterprise Deployment

Configure server migration for the bi_server1 and bi_server2 Managed Servers. With server migration configured, if a failure occurs, the bi_server1 Managed Server restarts on APPHOST2, and the bi_server2 Managed Server restarts on APPHOST1. For this configuration, the bi_server1 and bi_server2 servers listen on specific floating IPs that are failed over by WLS Server Migration.

Perform the steps in the following sections to configure server migration for the Managed Servers.

11.2 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.

Perform the following steps 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 using the following steps:

   a. Copy the leasing.ddl file that is located in the following directory that corresponds to the database version:

      ```
      WL_HOME/server/db/oracle/db_version
      ```

   b. Connect to the database as the leasing user.

   c. Run the leasing.ddl script in SQL*Plus:

      ```
      SQL> @copy_location/leasing.ddl;
      ```

11.3 Creating a Multi-Data Source Using the Administration Console

Create a multi-data source for the leasing table from the Oracle WebLogic Server Administration Console.

You create a data source for each of the Oracle RAC database instances during the process of setting up the multi-data source, both for these data sources and the global leasing multi-data source. When you create a data source:

- Ensure that it is a non-XA data source.
- The names of the multi-data sources are in the format of *<MultiDS>-rac0*, *<MultiDS>-rac1*, and so on.
- Use Oracle's Driver (Thin) Version 9.0.1, 9.2.0, 10, 11.
- Data sources do not require support for global transactions. Therefore, do *not* use any type of distributed transaction emulation/participation algorithm for the data source (do not choose the **Supports Global Transactions** option, or the **Logging Last Resource**, **Emulate Two-Phase Commit**, or **One-Phase Commit** suboptions), and specify a service name for the database.
- Target these data sources to the bi_cluster.
- Ensure that the initial connection pool capacity of the data sources is set to 0 (zero). To do this, select **Services** and **Data Sources**. In the Data Sources list, click

the name of the data source, and click the **Connection Pool** tab and enter **0** (zero) in the **Initial Capacity** field.

For additional recommendations for setting up a multi-data source for Oracle RAC, see "Considerations for High Availability Oracle Database Access" in *Oracle Fusion Middleware High Availability Guide*.

Perform the following steps to create a multi-data source:

1. In the Domain Structure window in the Administration Console, expand the **Services** node, then click **Data Sources**. The Summary of JDBC Data Sources page is displayed.
2. In the Change Center, click **Lock & Edit**.
3. Click **New**, then select **Multi Data Source**. The Create a New JDBC Multi Data Source page is displayed.
4. For **Name**, enter `leasing`.
5. For **JNDI Name**, enter `jdbc/leasing`.
6. For **Algorithm Type**, select **Failover** (the default).
7. Click **Next.**
8. On the Select Targets page, select **bi_cluster** as the target.
9. Click **Next**.
10. On the Select Data Source Type page, select **non-XA driver** (the default).
11. Click **Next**.
12. Click **Create a New Data Source**.
13. For **Name**, enter `leasing-rac0`. For **JNDI Name**, enter `jdbc/leasing-rac0`. For **Database Type**, select **Oracle**.

    Note: When creating the multi-data sources for the leasing table, enter names in the format of <*MultiDS*>-rac0, <*MultiDS*>-rac1, and so on.

14. Click **Next**.
15. For **Database Driver**, select **Oracle's Driver (Thin) for RAC Service-Instance connections; Versions:10 and later**.
16. Click **Next**.
17. Deselect **Supports Global Transactions**.
18. Click **Next**.
19. Enter the leasing schema details, as follows:
    - **Service Name:** Enter the service name of the database.
    - **Database name:** Enter the Instance Name for the first instance of the Oracle RAC database.
    - **Host Name:** Enter the name of the node that is running the database. For the Oracle RAC database, specify the first instance's VIP name or the node name as the host name.
    - **Port:** Enter the port number for the database (`1521`).

- **Database User Name:** Enter `leasing`.
- **Password:** Enter the leasing password.

20. Click **Next**.
21. Click **Test Configuration** and verify that the connection works.
22. Click **Next**.
23. On the Select Targets page, select **bi_cluster** as the target.
24. Click **Finish**.
25. Click **Create a New Data Source** for the second instance of the Oracle RAC database and target it to the bi_cluster, while repeating the steps for the second instance of the Oracle RAC database.
26. On the Add Data Sources page, add **leasing-rac0** and **leasing-rac1** to the datasource by moving them to the **Chosen** list.
27. Click **Finish**.
28. Click **Activate Changes**.

11.4 Enabling Host Name Verification Certificates

Create the appropriate certificates for host name verification between Node Manager and the Administration Server. This procedure is described in Section 10.3, "Enabling Host Name Verification Certificates for Node Manager." If you have not yet created these certificates, then perform the steps in this section to create certificates for host name verification between Node Manager and the Administration Server.

11.5 Editing the Node Manager Properties File

Edit the Node Manager properties file for the Node Managers in both nodes on which server migration is being configured. 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=eth0
NetMask=255.255.255.0
UseMACBroadcast=true
```

- **Interface:** This property specifies the interface name for the floating IP (for example, eth0).

  Do not specify the sub-interface, such as `eth0:1` or `eth0:2`. This interface is to be used without `:0` or `:1`. Node Manager scripts traverse the different :*X*-enabled IPs to determine which to add or remove. For example, the valid values in Linux environments are eth0, eth1, eth2, eth3, eth*n*, depending on the number of interfaces configured.
- **NetMask:** This property specifies the net mask for the interface for the floating IP. The net mask is the same as the net mask on the interface; 255.255.255.0 is used as an example in this document.
- **UseMACBroadcast:** This property specifies whether to use a node's MAC address when sending ARP packets, or in other words, whether to use the `-b` flag in the `arping` command.

Verify in the Node Manager output (the shell where Node Manager is started) that these properties in use. Otherwise, problems might occur during migration. The output is similar to the following:

```
...
StateCheckInterval=500
Interface=eth0
NetMask=255.255.255.0
...
```

> **Note:** The following steps are not required if the server properties (start properties) have been properly set and Node Manager can start the servers remotely.

1. Set the following property in the nodemanager.properties file:
   - **StartScriptEnabled:** Set this property to "true". This setting is required for Node Manager to start the Managed Servers using start scripts.
2. Start Node Manager on APPHOST1 and APPHOST2 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 might require different NetMask or Interface properties. In this case, specify individual parameters on a per-node basis using environment variables. For example, to use a different interface (eth3) in *HOSTn*, use the Interface environment variable as follows:
>
> ```
> HOSTn> export JAVA_OPTIONS=-DInterface=eth3
> ```
>
> Then, start Node Manager after the variable has been set in the shell.

11.6 Setting Environment and Superuser Privileges for the wlsifconfig.sh Script

Perform the following steps to set the environment and superuser privileges for the wlsifconfig.sh script:

1. Ensure that the PATH environment variable includes the files that are described in Table 11–1.

Table 11–1 Files Required for the PATH Environment Variable

File	Located in This Directory
wlsifconfig.sh	*ORACLE_BASE*/admin/*domain_name*/mserver/*domain_name*/bin/server_migration
wlscontrol.sh	*WL_HOME*/common/bin
nodemanager.domains	*WL_HOME*/common/nodemanager

2. Grant sudo configuration for the wlsifconfig.sh script.
   - Configure sudo to work without a password prompt.

- For security reasons, sudo must be restricted to the subset of commands that are required to run the wlsifconfig.sh script. For example, perform these steps to set the environment and superuser privileges for the wlsifconfig.sh script:

  a. Grant sudo privilege to the WebLogic user ("oracle") with no password restriction, and grant execute privilege on the /sbin/ifconfig and /sbin/arping binaries.

  b. Ensure that 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`:

  ```
  Defaults:oracle !requiretty
  oracle ALL=NOPASSWD: /sbin/ifconfig,/sbin/arping
  ```

  Note: Ask the system administrator for the sudo and system rights as appropriate to this step.

3. Restart the Node Manager so that the changes take effect.

11.7 Configuring Server Migration Targets

In this section, you configure server migration targets. Configuring Cluster Migration sets the DataSourceForAutomaticMigration property to true.

To configure migration in a cluster:

1. Log in to the Administration Console (http://*Host*:*Admin_Port*/console). Typically, *Admin_Port* is 7001 by default.
2. In the Domain Structure window, expand **Environment** and select **Clusters**. The Summary of Clusters page is displayed.
3. Click the cluster for which you want to configure migration (bi_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 **APPHOST1** and **APPHOST2**.
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. 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. For bi_server1, select APPHOST2. For bi_server2, select APPHOST1.

e. Select **Automatic Server Migration Enabled** and click **Save**.

   This action enables the Node Manager to start a failed server on the target node automatically.

f. Click **Activate Changes**.

11. Restart the Administration Server, Node Managers, Managed Servers, and the system components for which server migration has been configured.

> **Tip:** Click **Customize this table** in the Summary of Servers page, and move Current Machine from the Available window to the Chosen window to view the machine on which the server is running. These steps differ from the configuration in which the server is migrated automatically.

11.8 Testing the Server Migration

Perform the following steps to verify that server migration is working properly:

From APPHOST1:

1. Stop the bi_server1 Managed Server. To do this, run this command:

```
APPHOST1> 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:

```
APPHOST1> ps -ef | grep bi_server1
```

2. Watch the Node Manager console. You see a message that indicates that bi_server1's floating IP has been disabled.
3. Wait for Node Manager to try a second restart of bi_server1. It waits for a fence period of 30 seconds before trying this restart.
4. After Node Manager restarts the server, stop it again. Node Manager now logs a message that indicates that the server is not restarted again locally.

From APPHOST2:

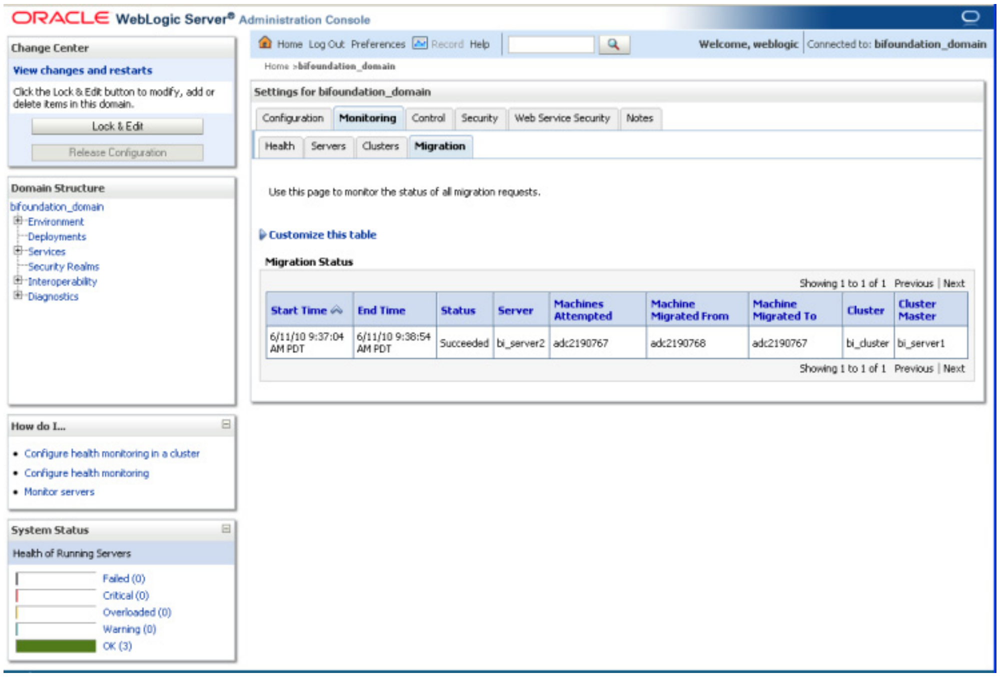
1. Watch the local Node Manager console. After 30 seconds since the last try to restart bi_server1 on node 1, Node Manager on node 2 prompts that the floating IP for bi_server1 is being brought up and that the server is being restarted in this node.
2. Access one of the applications (for example, BI Publisher) using the same IP.

Verification From the Administration Console

Perform the following steps to verify migration in the Administration Console:

1. Log in to the Administration Console.
2. Click **Domain** on the left console.
3. Click the **Monitoring** tab and the **Migration** tab.

   The Migration Status table provides information on the status of the migration, as shown in the following image:



Click the Lock & Edit button to modify, add or delete items in this domain.

Settings for bifoundation_domain

Use this page to monitor the status of all migration requests.

Migration Status

Showing 1 to 1 of 1 Previous | Next

Start Time	End Time	Status	Server	Machines Attempted	Machine Migrated From	Machine Migrated To	Cluster	Cluster Master
6/11/10 9:37:04 AM PDT	6/11/10 9:38:54 AM PDT	Succeeded	bi_server2	adc2190767	adc2190768	adc2190767	bi_cluster	bi_server1

Showing 1 to 1 of 1 Previous | Next

Note: After a server is migrated, to fail it back to its original node or computer, stop the Managed Server from the Administration Console and start it again. The appropriate Node Manager starts the Managed Server on the computer to which it was originally assigned.

12

Integrating an Enterprise Deployment with Oracle Identity Management

This chapter describes how to integrate Oracle Business Intelligence with Oracle Identity Management.

Before you perform the steps in this chapter, you must have successfully completed the installation and configuration steps described in both of the following:

- *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*
- The previous chapters of this guide

> **Important: Oracle strongly recommends that you read the *Oracle Fusion Middleware Release Notes* for any additional installation and deployment considerations before starting the setup process.**

This chapter contains the following topics:

- Section 12.1, "Configuring the Credential and Policy Store"
- Section 12.2, "Oracle Access Manager 10g Integration"
- Section 12.3, "Oracle Access Manager 11g Integration"
- Section 12.4, "Backing Up the Identity Management Configuration"

12.1 Configuring the Credential and Policy Store

This section contains the following topics:

- Section 12.1.1, "Overview of Credential and Policy Store Configuration"
- Section 12.1.2, "Configuring the Credential Store"
- Section 12.1.3, "Configuring the Policy Store"
- Section 12.1.4, "Reassociating Credentials and Policies"
- Section 12.1.5, "Refreshing User GUIDs After Identity Store Reassociation"

12.1.1 Overview of Credential and Policy Store Configuration

Oracle Fusion Middleware allows using different types of credentials 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 that are made on Managed Servers are not propagated to the Administration Server unless they use the same domain home. Because the Oracle Business Intelligence EDG topology uses different domain homes for the Administration Server and the Managed Server, 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 that are required to change the default store to Oracle Internet Directory LDAP for credentials or policies.

> **Note:** The back-end repository for the policy store and the credential store must use the same kind of LDAP server. To preserve this coherence, note that reassociating one store implies reassociating the other one; that is, the reassociation of both credential and the policy stores is accomplished as a unit using Oracle Enterprise Manager Fusion Middleware Control or the WLST command `reassociateSecurityStore`.

12.1.2 Configuring the Credential Store

This section explains how to configure the credential store and contains the following topics:

- Section 12.1.2.1, "Creating Users and Groups"
- Section 12.1.2.2, "Backing Up Configuration Files"
- Section 12.1.2.3, "Configuring the Identity Store to Use LDAP"
- Section 12.1.2.4, "Setting the Order of Providers"
- Section 12.1.2.5, "Moving the WebLogic Administrator to LDAP"

12.1.2.1 Creating Users and Groups

Create the users and groups that you need in Oracle Internet Directory, if you have not done so already. See *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory* and Chapter 3: Using Alternative Authentication Providers in *Oracle Fusion Middleware Security Guide for Oracle Business Intelligence Enterprise Edition* for more information.

12.1.2.2 Backing Up Configuration Files

To ensure that you have a copy of the latest version of files, 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.

12.1.2.3 Configuring the Identity Store to Use LDAP

Perform the following steps to configure the credential store to use LDAP by setting the proper authenticator using the Oracle WebLogic Server Administration Console:

1. Log in to the Administration Console.
2. Click the **Security Realms** link on the left navigation bar.
3. Click the **myrealm** default realm entry to configure it.
4. Open the **Providers** tab within the realm. Notice that there is a DefaultAuthenticator provider configured for the realm.
5. In the Change Center, click **Lock & Edit**.
6. Click **New** to add a new provider.
7. Enter a name for the provider, such as `OIDAuthenticator`.
8. Select the **OracleInternetDirectoryAuthenticator** type from the list of authenticators.
9. Click **OK**.
10. In the Providers screen, click the newly created authenticator.
11. Set the control flag to **SUFFICIENT**. This indicates that if a user can be authenticated successfully by this authenticator, then that authentication is accepted and any additional authenticators are not invoked. If the authentication fails, then it is passed to the next authenticator in the chain.

    Ensure that all subsequent authenticators also have their control flag set to SUFFICIENT. In particular, check the control flag for the DefaultAuthenticator and set it to SUFFICIENT if necessary.
12. Click **Save**.
13. Open the Provider Specific tab, then enter details that are specific to the LDAP server, as shown in Table 12–1.

Table 12–1 LDAP Server Details

Parameter	Value	Description
Host	For example: oid.mycompany.com	The host name of the LDAP server.
Port	For example: 636	The LDAP server port number.
Principal	For example: cn=orcladmin	The LDAP user DN used to connect to the LDAP server.
Credential	*your_password*	The password used to connect to the LDAP server.
SSL Enabled	Selected	Specifies whether SSL protocol is used when connecting to the LDAP server.
User Base DN	For example: cn=Users,dc=mycompany, dc=com	Specifies the DN under which the Users start.
Group Base DN	For example: cn=Groups,dc=mycompany, dc=com	Specifies the DN that points to the Groups node.
User Name Attribute	cn	The user name attribute.
Use Retrieved User Name as Principal	Selected	This option must be enabled.

14. Click **Save** when done.
15. Click **Activate Changes** to propagate the changes.
16. Restart the Administration Server and the Managed Servers.

12.1.2.4 Setting the Order of Providers

Reorder the OID Authenticator and Default Authenticator and ensure that the control flags for each authenticator is set as follows:

- OID LDAP Authenticator: SUFFICIENT
- Default Authenticator: SUFFICIENT

Restart the Administration Server, the Managed Servers, and the Oracle Business Intelligence system components.

12.1.2.5 Moving the WebLogic Administrator to LDAP

After LDAP has been configured, all users (including administrative users) must be LDAP users. This must be configured by the LDAP administrator. Create an administration group with the necessary users. For information about the required steps, see "Creating Users and Groups for Oracle Identity Manager" in *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*. Use "BIAdministrators" for the group name.

After this group is created, perform the following steps to update the role definition for the WLS Global Admin role in Oracle WebLogic Server:

1. Log in to the Administration Console.
2. Go to the location that defines the Admin role by selecting **Security Realms**, then the realm name, then **Role and Policies**, then **Global Roles**, then **Roles**, then **Admin**. Click the **View Role Conditions** link.

   By default, you can see that the Administrators group in Oracle Internet Directory defines who has the Admin role in Oracle WebLogic Server
3. Click **Add Conditions** to add a different group name (BIAdministrators). Then, delete the Administrators group, leaving the new one that you added.
4. Click **Save**.
5. After making this change, any members of the new group that you specified are authorized to administer Oracle WebLogic Server.

12.1.2.5.1 Updating the boot.properties File and Restarting the System

The boot.properties file for the Administration Server must be updated with the WebLogic admin user that was created in Oracle Internet Directory. Perform the following steps to update the boot.properties file:

1. On APPHOST1, change to the following directory:

   ```
   APPHOST1> cd ORACLE_BASE/admin/domain_name/aserver/
   domain_name/servers/AdminServer/security
   ```

2. Rename the existing boot.properties file:

   ```
   APPHOST1> 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=admin_user
   ```

```
password=admin_user_password
```

4. Save the file.
5. Stop and restart the Administration Server.

12.1.3 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 that are deployed in the domain can use. This section provides the steps to configure Oracle Internet Directory LDAP as the policy store for the Oracle Business Intelligence EDG topology.

To ensure proper access to the Oracle Internet Directory LDAP server directory that is used as a policy store, you must set a node in the server directory.

Perform the following steps as an Oracle Internet Directory administrator to create the appropriate node in the Oracle Internet Directory server:

1. Create an LDIF file (jpstestnode.ldif in this example), specifying the following DN and CN entries:

   ```
   dn: cn=jpsroot_bi,dc=mycompany,dc=com
   cn: jpsroot_bi
   objectclass: top
   objectclass: OrclContainer
   ```

   The DN of the root node (jpsroot_bi in the previous step) must be distinct from any other DN. 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 the Oracle Internet Directory server using the command `ldapadd`, as shown in the following example:

   ```
   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 shown in the following example:

   ```
   OIDHOST1> ORACLE_HOME/bin/ldapsearch -h ldap_host -p ldap_port -D cn=orcladmin
   -w password -b "cn=jpsroot_bi,dc=mycompany,dc=com" objectclass="orclContainer"
   ```

4. When using Oracle Internet Directory as the LDAP-Based policy store, run the oidstats.sql utility in the INFRADBHOST to generate database statistics for optimal database performance:

   ```
   OIDHOST1> connect ods/password
   OIDHOST1> @ORACLE_HOME/ldap/admin/oidstats.sql
   ```

   Note: The oidstats.sql utility needs to be run only once after the initial provisioning.

12.1.4 Reassociating Credentials and Policies

Perform the following steps to reassociate the policy and credential store with Oracle Internet Directory using the WLST `reassociateSecurityStore` command:

1. From APPHOST1, start the wlst shell:

```
APPHOST1> cd ORACLE_COMMON_HOME/common/bin
APPHOST1> ./wlst.sh
```

2. Connect to the WebLogic Administration Server using the wlst `connect` command, as follows:

   ```
   connect ("AdminUser", "AdminPassword", "t3://hostname:port")
   ```

   For example:

   ```
   connect ("weblogic", "password", "t3://ADMINVHN:7001")
   ```

3. Run the `reassociateSecurityStore` command, as follows:

   ```
   reassociateSecurityStore(domain="domainName", admin="cn=admin_user_name",
   password="orclPassword", ldapurl="ldap://LDAPHOST:LDAPPORT", servertype="OID",
   jpsroot="cn=jpsroot_bi")
   ```

   For example:

   ```
   wls:/bifoundation_domain/serverConfig>
   reassociateSecurityStore(domain="bifoundation_domain", admin="cn=orcladmin",
   password="password", ldapurl="ldap://oid.mycompany.com:389", servertype="OID",
   jpsroot="cn=jpsroot_bi,dc=mycompany,dc=com")
   ```

4. Restart the Administration Server after the command completes successfully.

> **Note:** For credential and policy changes to take effect, you must restart the servers in the domain.

12.1.5 Refreshing User GUIDs After Identity Store Reassociation

This section contains the following topics:

- Section 12.1.5.1, "About User GUIDs"
- Section 12.1.5.2, "About Refreshing GUIDs"
- Section 12.1.5.3, "Refreshing User GUIDs"

12.1.5.1 About User GUIDs

In Oracle Business Intelligence 11*g* Release 1 (11.1.1), users are recognized by their global unique identifiers (GUIDs), not by their names. GUIDs are identifiers that are completely unique for a given user. Using GUIDs to identify users provides a higher level of security, because it ensures that data and metadata is uniquely secured for a specific user, independent of the user name.

Oracle recommends that you follow these two best practices to ensure that GUIDs are consistently applied in each phase of the development to production lifecycle:

- Ensure that a fan-out replica of the identity store is used between development, test, and production systems, so that user GUIDs are consistent and identical across the complete development to production lifecycle. See "Setting Up Replication" in *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory* for further information about creating fan-out replicas.
- Wherever possible, secure access to data and metadata using application roles rather than individual users.

12.1.5.2 About Refreshing GUIDs

GUID refresh (also called GUID synchronization or GUID regeneration) updates any metadata references to user GUIDs in the Oracle BI repository and Oracle BI Presentation Catalog. During the GUID refresh process, each user name is looked up in the identity store. Then, all metadata references to the GUID associated with that user name are replaced with the GUID in the identity store.

GUID refresh might be required when Oracle Business Intelligence is reassociated with an identity store that has different GUIDs for the same users. This situation might occur when reassociating Oracle Business Intelligence with a different type of identity store and is usually a rare event.

Note that if Oracle best practices are not observed and Oracle Business Intelligence repository data is migrated between systems that have different GUIDs for the same users, GUID refresh is required for the system to function. This is not a recommended practice, because it raises the risk that data and metadata secured to one user (for example, John Smith, who left the company two weeks ago) becomes accessible to another user (for example, John Smith, who joined last week). Using application roles wherever possible and using GUIDs consistently across the full development production lifecycle prevents this problem from occurring.

12.1.5.3 Refreshing User GUIDs

To refresh user GUIDs, perform the following steps on APPHOST1 and APPHOST2. Note that GUID refresh must occur with only one node operating at a time.

1. Stop the Oracle BI Server and Presentation Services on all nodes except where you are refreshing the user GUIDs. For example:

   ```
   cd ORACLE_BASE/admin/instancen/bin
   ./opmnctl stopproc ias-component=coreapplication_obips1
   ./opmnctl stopproc ias-component=coreapplication_obis1
   ```

2. Update the `FMW_UPDATE_ROLE_AND_USER_REF_GUIDS` parameter in the NQSConfig.INI file using the following steps:

   a. Open the NQSConfig.INI file for editing in the following directory:

      ```
      ORACLE_INSTANCE/config/OracleBIServerComponent/coreapplication_obisn
      ```

   b. Locate the `FMW_UPDATE_ROLE_AND_USER_REF_GUIDS` parameter and set it to `YES`, as follows:

      ```
      FMW_UPDATE_ROLE_AND_USER_REF_GUIDS = YES;
      ```

   c. Save and close the file.

3. Update the Catalog element in the instanceconfig.xml file using the following steps:

   a. Open the instanceconfig.xml file for editing in the following directory:

      ```
      ORACLE_INSTANCE/config/OracleBIPresentationServicesComponent/
      coreapplication_obipsn
      ```

   b. Locate the Catalog element and update it as follows:

      ```
      <Catalog>
      <UpgradeAndExit>false</UpgradeAndExit>
      <UpdateAccountGUIDs>UpdateAndExit</UpdateAccountGUIDs>
      </Catalog>
      ```

c. Save and close the file.

4. On the node where you are refreshing the GUIDs, stop and start the Oracle BI Server and Presentation Services using the opmnctl command:

   ```
   cd ORACLE_BASE/admin/instancen/bin
   ./opmnctl stopproc ias-component=coreapplication_obips1
   ./opmnctl stopproc ias-component=coreapplication_obis1
   ./opmnctl startproc ias-component=coreapplication_obis1
   ```

   After you confirm that the Oracle BI Server is running, then start Presentation Services:

   ```
   ./opmnctl startproc ias-component=coreapplication_obips1
   ```

5. Set the FMW_UPDATE_ROLE_AND_USER_REF_GUIDS parameter in the NQSConfig.INI file back to NO.

   Important: You must perform this step to ensure that the system is secure.

6. Update the Catalog element in the instanceconfig.xml file to remove the UpdateAccount GUIDs entry.

7. Restart the Oracle Business Intelligence system components using the opmnctl command:

   ```
   cd ORACLE_BASE/admin/instancen/bin
   ./opmnctl stopall
   ./opmnctl startall
   ```

12.2 Oracle Access Manager 10g Integration

This section describes how to configure Oracle Access Manager 10*g* as a single sign-on solution for the Oracle Business Intelligence topology.

This section contains the following topics:

- Section 12.2.1, "About Oracle Access Manager Integration"
- Section 12.2.2, "Using the Oracle Access Manager Configuration Tool"
- Section 12.2.3, "Updating the Host Identifier"
- Section 12.2.4, "Updating the WebGate Profile"
- Section 12.2.5, "Installing and Configuring WebGate"
- Section 12.2.6, "Configuring IP Validation for WebGate"
- Section 12.2.7, "Setting Up WebLogic Authenticators"
- Section 12.2.8, "Configuring Applications"

12.2.1 About Oracle Access Manager Integration

The instructions for Oracle Access Manager 10*g* assume an existing Oracle Access Manager installation, complete with Access Managers and a policy that protects the Policy manager. For more information about installing and configuring an Oracle Access Manager installation, see *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*.

The configuration that is described in this chapter includes a directory service such as Oracle Internet Directory, either as a standalone component or as part of an Oracle Virtual Directory configuration. This section provides the necessary steps for configuring the Oracle Business Intelligence installation with Oracle Internet Directory.

In addition, the Oracle Access Manager installation should have its own web server configured with WebGate. This section also provides steps for using the Oracle Access Manager web server as a delegated authentication server.

12.2.2 Using the Oracle Access Manager Configuration Tool

This section explains how to use the Oracle Access Manager Configuration Tool and contains the following topics:

- Section 12.2.2.1, "About the Oracle Access Manager Configuration Tool"
- Section 12.2.2.2, "Collecting Information for the Oracle Access Manager Configuration Tool"
- Section 12.2.2.3, "Running the Oracle Access Manager Configuration Tool"
- Section 12.2.2.4, "Verifying Successful Creation of the Policy Domain and AccessGate"

12.2.2.1 About the Oracle Access Manager Configuration Tool

The Oracle Access Manager Configuration Tool (oamcfgtool) starts a series of scripts and sets up the required policies. It requires various parameters as inputs. Specifically, the tool creates the following:

- A Form Authentication scheme in Oracle Access Manager
- Policies to enable authentication in Oracle WebLogic Server
- A WebGate entry in Oracle Access Manager to enable Oracle HTTP Server WebGates (from the Web tier) to protect the configured application
- A Host Identifier, depending on the scenario chosen (a default host identifier is used, if not provided)
- Policies to protect and unprotect the application-specific URL

12.2.2.2 Collecting Information for the Oracle Access Manager Configuration Tool

Collect or prepare the following information before running the Oracle Access Manager Configuration Tool:

- Password: Create a secure password. This is used as the password for the WebGate installation performed later.
- LDAP Host: The host name of the Directory Server or load balancer address, for HA/EDG configurations.
- LDAP Port: The port number of the Directory Server.
- LDAP USER DN: The DN of the LDAP administrator user (for example, "cn=orcladmin").
- LDAP password: The password of the LDAP administrator user.
- OAM_AA_HOST: The host name of the Oracle Access Manager instance.
- OAM_AA_PORT: The Oracle Access Manager port number.

12.2.2.3 Running the Oracle Access Manager Configuration Tool

The Oracle Access Manager Configuration Tool is located in the following directory:

```
MW_HOME/oracle_common/modules/oracle.oamprovider_11.1.1
```

You can run the tool from any computer with the required installation files. In this case, you run it from APPHOST1.

> **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.

Set the JAVA_HOME value before running the tool using the following command:

```
export JAVA_HOME=$MW_HOME/jrockit_160_05_R27.6.2-20
```

Run the Oracle Access Manager Configuration Tool, as follows (all on a single line):

```
$JAVA_HOME/bin/java -jar oamcfgtool.jar mode=CREATE
app_domain="bifoundation_domain" protected_uris="$PROTECTED_URI_LIST"
public_uris="$PUBLIC_URI_LIST" ldap_host="oid.mycompany.com" ldap_port=389
ldap_userdn="cn=LDAP_admin_user_name"
ldap_userpassword=LDAP_admin_user_password oam_aaa_host=OAMHOST1
oam_aaa_port=OAMPORT1 oam_aaa_mode=simple
```

For `$PROTECTED_URI_LIST`, use:

```
"/bicontent,/mapviewer,/em/.../*,/console/.../*,/aps,/calcmgr,/hr,
/workspace,/analytics/saw.dll,/xmlpserver,/ui,/em,/console,/ui/adfAuthentication,/
mobile,/mobile/.../*"
```

For `$PUBLIC_URI_LIST`, use:

```
"/analytics,/analytics/saw.dll/wsdl,/analytics-ws/saw.dll,/xmlpserver/services,
/xmlpserver/report_service,/xmlpserver/ReportTemplateService.xls,
/xmlpserver/Guest,/ui/do/logout,/ui/images,/biservices"
```

You are prompted for the app_agent_password.

> **Note:** If additional URLs must be protected later, then run the Oracle Access Manager Configuration Tool again using the same app_domain. Ensure that you include all the URLs that must be protected, not just the new ones.

12.2.2.4 Verifying Successful Creation of the Policy Domain and AccessGate

This section describes how to validate that the Policy Domain and AccessGate were created successfully.

Verifying the Policy Domain

Perform the following steps to verify the policy domain:

1. Log on to Oracle Access Manager at:

   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 is displayed, including the domain that you just created.
4. Click the link to the policy domain that you just created. The General area of the domain is displayed.

5. Click the **Resources** tab. The URIs that you specified are displayed. You can also click other tabs to view other settings.

Verifying the AccessGate Configuration

Perform the following steps to verify the AccessGate configuration:

1. Click the **Access System Console** link on the top right. Note that this link toggles between Access System Console and Policy Manager when you click it.
2. Click the **Access System Configuration** tab.
3. Click the **AccessGate Configuration** link in the left pane.
4. Enter `bifoundation_domain` as the search criterion (or another substring in the app_domain), then click **Go**.

   The AccessGate for the domain that you just created is displayed. This result has the suffix _AG (for example, bifoundation_domain_AG).
5. Click the AccessGate for the domain to see details.

12.2.3 Updating the Host Identifier

The Oracle Access Manager Configuration Tool uses the value of the app_domain parameter to create a host identifier for the policy domain. This host identifier must be updated with all the host name variations for the host so that the configuration works correctly.

Perform the following steps to update the host identifier that is created by the Oracle Access Manager Configuration Tool:

1. Navigate to the Access System Console by entering the following URL in a web browser:

   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. When prompted for a user name and password, log in as an administrator. Click **OK**.
3. On the Access System main page, click the **Access System Console** link.
4. On the Access System Console page, click the **Access System Configuration** tab.
5. On the Access System Configuration page, click **Host Identifiers** on the bottom left.
6. On the List all host identifiers page, click the host identifier that was created by the Oracle Access Manager Configuration Tool. For example, select bifoundation_ domain.
7. On the Host Identifier Details page, click **Modify**.
8. On the Modifying host identifier page, add all the possible host name variations for the host. Click the plus and minus symbols to add or delete fields as necessary.

   The Preferred HTTP Host value used in the Access System Configuration must be added as one of the host name variations. For example:

   ```
   bifoundation_domain, webhost1.mycompany.com:7777, webhost2.mycompany.com:7777,
   APPHOST1VHN1.mycompany.com:9704, APPHOST2VHN1.mycompany.com:9704,
   ADMIN.mycompany.com:80, ADMINVHN.mycompany.com:7001, APPHOST1VHN1:9704,
   APPHOST2VHN1:9704, ADMINVHN:7001
   ```

9. Select **Update Cache** and click **Save**.

   The following message is displayed: "Updating the cache at this point will flush all the cache in the system. Are you sure?"

   Click **OK** to finish saving the configuration changes.

10. Verify the changes on the Host Identifier Details page.

12.2.4 Updating the WebGate Profile

The Oracle Access Manager 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 of these attributes must be updated with the correct values for the configuration to work.

Perform the following steps to update the WebGate profile that was created by the Oracle Access Manager Configuration Tool:

1. Navigate to the Access System Console by entering the following URL in a web browser:

   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. When prompted for a user name and password, log in as an administrator. Click **OK**.

3. On the Access System main page, click the **Access System Console** link.

4. On the Access System Console page, click the **Access System Configuration** tab to display the AccessGate Search page.

5. Enter the appropriate search criteria and click **Go** to display a list of AccessGates.

6. Select the AccessGate that was created by the Oracle Access Manager Configuration Tool. For example: bifoundation_domain_AG

7. On the AccessGate Details page, select **Modify** to display the Modify AccessGate page.

8. On the Modify AccessGate page, update the following:

   - **Hostname:** Update the host name with the name of the computer on which WebGate is running. For example: webhost1.mycompany.com

   - **Preferred HTTP Host:** Update the Preferred_HTTP_Host with one of the host name variations that is specified in the previous section. For example: webhost1.mycompany.com:7777

   - **Primary HTTP Cookie Domain:** Update the Primary HTTP Cookie Domain with the Domain suffix or the host identifier. For example: mycompany.com

   - **Port:** Update the port with the port number on which WebGate is running. For example: 7777

   - **Maximum Connections:** Set to 4.

9. Click **Save**, then click **OK** to confirm.

10. Verify the values that are displayed on the Details for AccessGate page to confirm that the updates were successful.

12.2.5 Installing and Configuring WebGate

WebGate must be installed on each of the WEBHOST*n* computers to secure the Web tier. Perform the following steps to install and configure WebGate:

1. Launch the WebGate installer using the following command:

   ```
   ./Oracle_Access_Manager10_1_4_3_0_linux_OHS11g_WebGate -gui
   ```

2. The Welcome screen is displayed. Click **Next**.
3. In the Customer Information screen, enter the user name and user group under which the Web server is running. Click **Next** to continue.
4. In the installation target screen, specify the directory where WebGate is installed. Click **Next** to continue.
5. In the installation summary screen, click **Next**.
6. Download the required GCC runtime libraries for WebGate as instructed in the WebGate configuration screen, and use **Browse** to point to their location on the local computer. Click **Next** to continue.
7. The installer now creates the required artifacts. After that process is complete, click **Next** to continue.
8. In the transport security mode screen, select the same mode that was configured for the BI Access Gate (for example, **Simple**) and click **Next** to continue.

   > **Note:** When integrating with Oracle Identity Management, use the transport mode that is currently in use by the Oracle Identity Management servers. For example, Open, Simple, or Cert.

9. In the WebGate Configuration screen, provide the details of the Access Server that are used. You must provide the following information:

   - WebGate ID, as provided when the Oracle Access Manager Configuration Tool was executed
   - Password for WebGate
   - Access Server ID, as reported by the Oracle Access Manager Access Server configuration
   - Access Server host name, as reported by the Oracle Access Manager Access Server configuration
   - Access Server port number, as reported by the Oracle Access Manager Access Server configuration
   - Global Access Protocol Pass Phrase

   You can obtain these details from the Oracle Access Manager administrator. Click **Next** to continue.

10. In the Configure Web Server screen, click **Yes** to automatically update the Web server. Click **Next** to continue.
11. In the next Configure Web Server screen, specify the full path of the directory that contains the httpd.conf file. Click **Next** to continue.
12. In the next Configure Web Server page, a message informs you that the web server configuration has been modified for WebGate. Click **Yes** to confirm.

13. Stop and start the web server for the configuration updates to take effect. Click **Next** to continue.
14. In the next Configure Web Server screen, a message about SSL is displayed. Click **Next** to continue.
15. In the next Configure Web Server screen, a message with the location of the document that has information about the rest of the product setup and web server configuration is displayed. Choose **No** and click **Next** to continue.
16. The final Configure Web Server screen is displayed with a message to manually launch a browser and open the HTML document for further information on configuring the web server. Click **Next** to continue.
17. The Oracle COREid Readme screen is displayed. Review the information on the screen and click **Next** to continue.
18. A message is displayed, providing details of the installation and informing you that the installation was successful.

12.2.6 Configuring IP Validation for WebGate

IP Validation determines if a client's IP address is the same as the IP address that is stored in the ObSSOCookie that is generated for single sign-on. IP Validation can cause issues in systems using load balancer devices that are 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. Perform the following steps to configure the 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 *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 log in as an administrator.
3. On the Access System Console main page, click **Access System Configuration**, and click the **Access Gate Configuration** link on the left pane to display the AccessGates Search page.
4. Enter the appropriate search criteria and click **Go** to display a list of AccessGates.
5. Select the AccessGate that is 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 that is used to front-end the deployment.
8. Click **Save** at the bottom of the page.

12.2.7 Setting Up WebLogic Authenticators

The instructions in this section assume that you have already configured the LDAP Authenticators.

This section contains the following topics:

- Section 12.2.7.1, "Setting Up the Oracle Access Manager ID Asserter"

- Section 12.2.7.2, "Setting the Order of Providers"

12.2.7.1 Setting Up the Oracle Access Manager ID Asserter

Perform the following steps to set up the Oracle Access Manager ID Asserter:

1. Log in to the Administration Console.
2. In the Change Center, click **Lock & Edit**.
3. Navigate to SecurityRealms\myrealm\Providers.
4. Click **New** and select **OAM Identity Asserter** from the drop-down menu.
5. Name the asserter (for example: OAM ID Asserter) and click **OK**.
6. Click the newly added asserter to see the configuration screen for OAM Identity Asserter.
7. Set the control flag to **REQUIRED** and click **Save**.
8. Open the Provider Specific tab to configure the following required settings:
   - **Primary Access Server:** Provide the Oracle Access Manager server endpoint information in *HOST*:*PORT* format.
   - **AccessGate Name:** Provide the name of the AccessGate (for example, bifoundation_domain_AG).
   - **AccessGate password:** Provide the password for the AccessGate.
9. Click **Save** when done
10. Click **Activate Changes** to propagate the changes.
11. Restart the Administration Server and the Managed Servers.

12.2.7.2 Setting the Order of Providers

Reorder the Oracle Access Manager Identity Asserter, Oracle Internet Directory Authenticator, and Default Authenticator by ensuring that the control flag for each authenticator is set, as follows:

- OAM Identity Asserter: REQUIRED
- OID LDAP Authenticator: SUFFICIENT
- Default Authenticator: SUFFICIENT

Then, restart the Administration Server, the Managed Servers, and the Oracle Business Intelligence system components.

12.2.8 Configuring Applications

This section explains how to configure applications, and contains the following topics:

- Section 12.2.8.1, "Enabling SSO/Oracle Access Manager for Oracle BI EE"
- Section 12.2.8.2, "Enabling SSO and Oracle Access Manager for BI Publisher"
- Section 12.2.8.3, "Enabling SSO/Oracle Access Manager for Oracle BI Search"
- Section 12.2.8.4, "Enabling SSO/Oracle Access Manager for Oracle RTD"

12.2.8.1 Enabling SSO/Oracle Access Manager for Oracle BI EE

Perform the following steps to enable SSO and Oracle Access Manager for Oracle BI EE:

1. Log in to Fusion Middleware Control.
2. Go to **Business Intelligence > coreapplication > Security**.
3. Click **Lock and Edit Configuration**.
4. Choose **Enable SSO** and select **Oracle Access Manager** for SSO Provider.
5. Configure the login/logout information for the Oracle BI Presentation Services processes by entering the logon and logoff URLs in the following fields:
   - **The SSO Provider Logon URL:** http://*OAM_host*:*OAM_port*/oamsso/login.html
   - **The SSO Provider Logoff URL:** http://*OAM_host*:*OAM_port*/access/oblix/lang/en-us/logout.html
6. Click **Apply**.
7. Click **Activate Changes**.
8. Restart all Oracle Business Intelligence system components using opmnctl or Fusion Middleware Control.

12.2.8.2 Enabling SSO and Oracle Access Manager for BI Publisher

Perform the following steps to enable SSO and Oracle Access Manager for BI Publisher:

1. In BI Publisher, go to the **Administration > Security Configuration** page to enable SSO.
2. On the Security Configuration Page, provide the following information in the Single Sign-On section:
   a. Select **Use Single Sign-On**.
   b. For **Single Sign-On Type**, select **Oracle Access Manager**.
   c. For **Single Sign-Off URL**, enter a URL of the following format:

   ```
   http://OAM_host:OAM_port/access/oblix/lang/en-us/logout.html
   ```
3. Click **Apply**.
4. Restart the **bipublisher** application from the Administration Console.

12.2.8.3 Enabling SSO/Oracle Access Manager for Oracle BI Search

Perform the following steps to enable SSO and Oracle Access Manager for Oracle BI Search:

1. Open the BISearchConfig.properties file for editing in the following directory:

   ```
   DOMAIN_HOME/config/fmwconfig/biinstances/coreapplication/
   ```
2. Set the value of BIServerSSOUrl to the following:

   https://bi.mycompany.com/analytics
3. Save and close the file.

12.2.8.4 Enabling SSO/Oracle Access Manager for Oracle RTD

This section provides information about Oracle RTD configuration with Oracle Access Manager.

This section contains the following topic:

- Section 12.2.8.4.1, "Oracle RTD and Oracle Access Manager 10g Logout Guidelines"

12.2.8.4.1 Oracle RTD and Oracle Access Manager 10*g* Logout Guidelines For Oracle RTD to comply with Oracle Access Manager logout guidelines (in particular, invoking a logout through /adfAuthentication?logout=true&end_url=/ui/do/logout), integration with Oracle Access Manager 10*g* requires additional WebGate configuration 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 Application Security Guide*.

12.3 Oracle Access Manager 11*g* Integration

This section describes how to configure Oracle Access Manager 11*g* as the single sign-on solution for the Oracle Business Intelligence Enterprise Deployment topology.

This section contains the following sections:

- Section 12.3.1, "Overview of Oracle Access Manager Integration"
- Section 12.3.2, "Prerequisites for Oracle Access Manager"
- Section 12.3.3, "Install WebGate"
- Section 12.3.4, "Register the WebGate Agent"
- Section 12.3.5, "Configuring IP Validation for WebGate"
- Section 12.3.6, "Setting Up the WebLogic Authenticators"
- Section 12.3.7, "Configuring Applications"

12.3.1 Overview of Oracle Access Manager Integration

Oracle Access Manager is the recommended single sign-on solution for Oracle Fusion Middleware 11*g* Release 1. For more information on installing and configuring an Oracle Access Manager installation, see *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*. This section explains the procedure for configuring the Oracle Business Intelligence installation with an existing Oracle Access Manager 11*g* installation and the underlying directory service. Oracle recommends using either Oracle Internet Directory, Oracle Virtual Directory, or both of these directory services.

> **Note:** The Oracle Business Intelligence topology that is described in this guide uses a Single Sign-On configuration where both the Oracle Business Intelligence system and the Single Sign-On system are in the same network domain (mycompany.com). For a multi-domain configuration, refer to the required configuration steps in Chapter 11, "Introduction to Single Sign-On with Oracle Access Manager 11*g*," in *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager with Oracle Security Token Service*.

12.3.2 Prerequisites for Oracle Access Manager

The setup for Oracle Access Manager assumes an existing Oracle Access Manager installation that is complete with Access Managers and a policy that is protecting the Policy Manager. For more information on installing and configuring Oracle Access Manager, see *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*. This setup includes a directory service such as Oracle Internet Directory, either standalone or as part of an Oracle Virtual Directory configuration. This chapter provides the necessary steps for configuring the Oracle Business Intelligence installation with either Oracle Internet Directory or Oracle Virtual Directory.

In addition, the Oracle Access Manager installation must have its own web server that is configured with a WebGate. This section also provides the steps for using the Oracle Access Manager web server as a delegated authentication server.

12.3.3 Install WebGate

You must install a WebGate on each of the WEBHOST computers where an HTTP Server has already been installed. Repeat Section 12.3.3 and Section 12.3.4 for each WEBHOST in the deployment environment.

12.3.3.1 Installing GCC Libraries

You must download and install third-party GCC libraries on the computer before installing WebGate.

You can download the appropriate GCC library from the following third-party web site:

`http://gcc.gnu.org/`

For Linux 32-bit, the required libraries are libgcc_s.so.1 and libstdc++.so.5 with a version number of 3.3.2. Table 12–2 lists the versions of GCC third-party libraries for Linux and Solaris.

Table 12–2 Versions of GCC Third-Party Libraries for Linux and Solaris

Operating System	Architecture	GCC Libraries	Required Library Version
Linux 32-bit	x86	libgcc_s.so.1 libstdc++.so.5	3.3.2
Linux 64-bit	x64	libgcc_s.so.1 libstdc++.so.6	3.4.6
Solaris 64-bit	SPARC	libgcc_s.so.1 libstdc++.so.5	3.3.2

12.3.3.2 Installing WebGate

This section describes the procedures for installing WebGate.

Launching the Installer

The Installer program for Oracle HTTP Server 11*g* WebGate for Oracle Access Manager is included in the webgate.zip file.

Perform the following steps to start the installation wizard:

1. Extract the contents of the webgate.zip file to a directory. By default, this directory is namedwebgate.

2. Move to the Disk1 directory under the webgate folder.
3. Start the installer using the following command:

   ```
   $ ./runInstaller -jreLoc WebTier_Home/jdk
   ```

   After the installer starts, the Welcome screen is displayed.

Installation Flow and Procedure

If you need additional help with any of the installation screens, then click **Help** to access the online help.

Perform the following steps to install Oracle HTTP Server 11*g* WebGate for Oracle Access Manager:

1. In the Welcome screen, click **Next**.
2. In the Prerequisite Checks screen, click **Next**.
3. In the Specify Installation Location screen, specify the Middleware Home and Oracle Home locations. You can use the default location, or choose another location.

   > **Note:** The Middleware home contains an Oracle home for Oracle Web Tier.

   Click **Next**.
4. In the Specify GCC Library screen, specify the directory that contains the GCC libraries, and click **Next**.
5. In the Installation Summary screen, verify the information on this screen and click **Install** to begin the installation.
6. In the Installation Progress screen, you might be prompted to run the *ORACLE_HOME*/oracleRoot.sh script to configure the proper file and directory permissions.

   Click **Next** to continue.
7. In the Installation Complete screen, click **Finish** to exit the installer.

12.3.3.3 Post-Installation Steps

Perform the following steps after installing Oracle HTTP Server 11*g* WebGate for Oracle Access Manager:

1. Move to the following directory under the Oracle home for WebGate:

   ```
   $ cd Webgate_Home/webgate/ohs/tools/deployWebGate
   ```

2. On the command line, run the following command to copy the required bits of agent from the *Webgate_Home* directory to the WebGate Instance location:

   ```
   $ ./deployWebGateInstance.sh -w Webgate_Instance_Directory
   -oh Webgate_Oracle_Home
   ```

   where *Webgate_Oracle_Home* is the directory where you have installed Oracle HTTP Server WebGate and created as the Oracle home for WebGate, as in the following example:

   ```
   MW_HOME/Oracle_OAMWebGate1
   ```

The *Webgate_Instance_Directory* is the location of WebGate Instance Home, which is the same as the Instance Home of Oracle HTTP Server, as in the following example:

```
MW_HOME/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 the present working directory, move up one directory level:

```
$ cd Webgate_Home/webgate/ohs/tools/setup/InstallTools
```

5. On the command line, run the following command to copy the apache_webgate.template from the *Webgate_Home* directory to the WebGate Instance location (renamed to webgate.conf) and update the httpd.conf file to add one line to include the name of webgate.conf:

```
$ ./EditHttpConf -w Webgate_Instance_Directory [-oh Webgate_Oracle_Home]
[-o output_file]
```

Note: The -oh *WebGate_Oracle_Home* and -o *output_file* parameters are optional.

---

where *WebGate_Oracle_Home* is the directory where you have installed Oracle HTTP Server WebGate for Oracle Access Manager and created as the Oracle Home for WebGate, as in the following example:

```
MW_HOME/Oracle_OAMWebGate1
```

The *Webgate_Instance_Directory* is the location of WebGate Instance Home, which is same as the Instance Home of Oracle HTTP Server, as in the following example:

```
MW_HOME/ORACLE_BASE/admin/webn/config/OHS/ohsn
```

The *output_file* is the name of the temporary output file that is used by the tool, as in the following example:

```
Edithttpconf.log
```

12.3.4 Register the WebGate Agent

This section describes the procedures for registering the WebGate Agent and contains the following topics:

- Section 12.3.4.1, "The RREG Tool"
- Section 12.3.4.2, "Updating the OAM11gRequest File"
- Section 12.3.4.3, "Running the oamreg Tool"
- Section 12.3.4.4, "Copying Access Files to WEBHOSTs"

12.3.4.1 The RREG Tool

The RREG tool is part of the Oracle Access Manager 11*g* installation. If it is not already available, then perform the following steps to extract it:

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:

```
/analytics/saw.dll
/bicontent
/xmlpserver
/ui
/mapviewer
/bicomposer
/bisearch
/em
/em/…/*
/console
/console/…/*
/calcmgr
/hr
/workspace
/ui/adfAuthentication
/mobile
/mobile/.../*
/bioffice
```

where "/.../*" implies all resources under the base url context.

The list of public resources is:

```
/analytics
/analytics/saw.dll/wsdl
/analytics-ws/saw.dll
/ui/do/logout
/xmlpserver/services
/xmlpserver/report_service
/xmlpserver/ReportTemplateService.xls
/xmlpserver/Guest
/biservices
/bioffice/services/saw?WSDL
/hr/services
/aps
/aps/JAPI
```

```
/aps/Essbase
/hr/modules/com/hyperion/reporting/web/repository/HRRepositoryXML.jsp
/hr/modules/com/hyperion/reporting/web/images
/ui/images/*
```

The list of excluded resources is:

```
/rtis
/rtis/.../*
/schema
/schema/.../*
/ws
/ws/.../*
/wsm-pm
/wsm-pm/.../*
```

12.3.4.2 Updating the OAM11gRequest File

In the *RREG_Home*/input directory, there is a template file named OAM11GRequest.xml. Copy this template to a new file called BIOAM11GRequest.xml and edit it to create the policies for the Oracle Business Intelligence installation. After editing, the file looks as follows.

> **Note:** Replace $$webtierhost$$, $$oamadminserverport$$, $$oamhost$$, and *load_balancer_source_IP* with their respective values in the 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>$$webtierhost$$_bi</hostIdentifier>
    <agentName>$$webtierhost$$_bi</agentName>
    <applicationDomain>$$webtierhost$$_bi</applicationDomain>
    <cachePragmaHeader>private</cachePragmaHeader>
    <cacheControlHeader>private</cacheControlHeader>
    <ipValidation>1</ipValidation>
    <logOutUrls>
        <url>/oamsso/logout.html</url>
    </logOutUrls>
    <protectedResourcesList>
        <resource>/analytics/saw.dll</resource>
        <resource>/bicontent</resource>
        <resource>/xmlpserver</resource>
        <resource>/ui</resource>
        <resource>/mapviewer</resource>
        <resource>/bicomposer</resource>
        <resource>/bisearch</resource>
        <resource>/em</resource>
        <resource>/em/…/*</resource>
        <resource>/bioffice</resource>
        <resource>/console</resource>
        <resource>/console/…/*</resource>
        <resource>/mobile</resource>
        <resource>/mobile/.../*</resource>
        <resource>/calcmgr</resource>
        <resource>/hr</resource>
        <resource>/workspace</resource>
        <resource>/ui/adfAuthentication</resource>
    </protectedResourcesList>
    <publicResourcesList>
        <resource>/analytics</resource>
        <resource>/analytics/saw.dll/wsdl</resource>
        <resource>/aps</resource>
        <resource>/ui/do/logout</resource>
        <resource>/xmlpserver/services</resource>
        <resource>/xmlpserver/report_service</resource>
        <resource>/bioffice/services/saw?WSDL</resource>
        <resource>/hr/services</resource>
        <resource>/aps/JAPI</resource>
        <resource>/aps/Essbase</resource>
        <resource>/hr/modules/com/hyperion/reporting/web/repository/HRRepositoryXM
L.jsp</resource>
        <resource>/hr/modules/com/hyperion/reporting/web/images</resource>
        <resource>/xmlpserver/ReportTemplateService.xls</resource>
        <resource>/xmlpserver/Guest</resource>
        <resource>/biservices</resource>
        <resource>/ui/images/*</resource>
        <resource>/analytics-ws/saw.dll</resource>
    </publicResourcesList>
    <excludedResourcesList>
        <resource>/rtis</resource>
        <resource>/rtis/.../*</resource>
        <resource>/schema</resource>
        <resource>/schema/.../*</resource>
        <resource>/ws</resource>
        <resource>/ws/.../*</resource>
        <resource>/wsm-pm</resource>
        <resource>/wsm-pm/.../*</resource>
    </excludedResourcesList>
</OAM11GRegRequest>
```

12.3.4.3 Running the oamreg Tool

Run the oamreg tool using the following command:

```
$ RREG_Home/bin/oamreg.sh inband RREG_Home/input/BIOAM11gRequest.xml
```

Note that the JAVA_HOME operating system environment variable must be set to jdk6 for this command to work.

The output looks similar to the following:

```
------------------------------------------------
Welcome to OAM Remote Registration Tool!
Parameters passed to the registration tool are:
Mode: inband
Filename: /u01/oim/oim_home/oam/server/rreg/client/rreg/input/BIOAM11GRequest.xml
Enter admin username: oamadmin_user
Username: oamadmin_user
Enter admin password: my_password
Do you want to enter a Webgate password?(y/n):
```

```
y
Enter webgate password: my_password
Enter webgate password again: my_password
Password accepted. Proceeding to register..
Nov 9, 2011 6:48:44 PM
oracle.security.am.engines.rreg.client.handlers.request.OAM11GRequestHandler
getWebgatePassword
INFO: Passwords matched and accepted.
Do you want to import an URIs file?(y/n):
n
----------------------------------------
Request summary:
OAM11G Agent Name:WEBHOST_bi
URL String:WEBHOST_bi
Registering in Mode:inband
Your registration request is being been sent to the Admin server at:
http://oamserver.mycompany.com:OAM_ADMINSERVER_PORT
----------------------------------------
Inband registration process completed successfully! Output artifacts are created
in the output folder.
```

12.3.4.4 Copying Access Files to WEBHOSTs

In OPEN mode, the following two files are generated in the *OAM_REG_HOME*`/output/$$webtierhost$$_bi` directory:

- ObAccessClient.xml
- cwallet.sso

Copy these files to the webgate instance (*Webgate_Instance_Home*`/config/OHS/ohsN/webgate/config/`) location on WEBHOST1 and WEBHOST2.

In SIMPLE mode, copy the following files from the *OAM_REG_HOME*`/output/$$webtierhost$$_bi` directory to the *Webgate_Instance_Home*`/webgate/config` directory on WEBHOST1 and WEBHOST2:

- ObAccessClient.xml
- cwallet.sso
- password.xml

In addition, copy the following files from the *OAM_REG_HOME*`/output/$$webtierhost$$_bi` directory to the *Webgate_Instance_Home*`/config/OHS/ohsN/webgate/config/simple` directory on WEBHOST1 and WEBHOST2:

- aaa_key.pem
- aaa_cert.pem

> **Note:** When integrating with Oracle Identity Management, use the transport mode that is currently in use by the Oracle Identity Management servers. For example, Open, Simple, or Cert.

After you copy the access files to WEBHOST1 and WEBHOST2, you must restart the Oracle HTTP Server instances for the changes to take effect.

12.3.5 Configuring IP Validation for WebGate

IP Validation determines if a client's IP address is the same as the IP address that is stored in the ObSSOCookie that is generated for single sign-on. IP Validation can cause issues in systems using load balancer devices that are configured to perform IP termination, or when the authenticating WebGate is front-ended by a different load balancer from the one that is front-ending the enterprise deployment. Perform the following steps to configure the load balancer so that it is not validated in these cases:

1. Go to the Oracle Access Manager 11*g* Console using the following URL:

   ```
   http://hostname:port/oamconsole
   ```

2. Log in as the Oracle Access Manager 11*g* Administrator.
3. On the Welcome page, click the **System Configuration** tab.
4. In the Access Manager Settings section, expand the **SSO Agents** node. Then, double-click **OAM Agents** to display the OAM Agents Search page.
5. Enter the appropriate search criteria and click **Search** to display a list of OAM Agents.
6. Select the OAM Agent that is created by the Oracle Access Manager configuration tool.
7. In the **IP Validation Exception** field, enter the address of the load balancer that is used to front-end the deployment.
8. Click **Apply** at the top of the page.

12.3.6 Setting Up the WebLogic Authenticators

This section assumes that you have already configured the LDAP authenticator by following the steps in Section 12.1.2.3, "Configuring the Identity Store to Use LDAP." If you have not already created the LDAP authenticator, then do so before continuing with this section.

This section contains the following topics:

- Section 12.3.6.1, "Back Up Configuration Files"
- Section 12.3.6.2, "Setting Up the OAM ID Asserter"
- Section 12.3.6.3, "Setting the Order of Providers"

12.3.6.1 Back Up Configuration Files

To be safe, first back up the relevant configuration files:

```
ORACLE_BASE/admin/domain_name/aserver/domain_name/config/config.xml
ORACLE_BASE/admin/domain_name/aserver/domain_name/config/fmwconfig/jps-config.xml
ORACLE_BASE/admin/domain_name/aserver/domain_
name/config/fwmconfig/system-jazn-data.xml
```

In addition, back up the boot.properties file for the Administration Server.

12.3.6.2 Setting Up the OAM ID Asserter

Perform the following steps to set up the OAM ID Asserter:

1. Log into Weblogic Console using the following URL:

   ```
   http://ADMINVHN.mycompany.com:7001/console
   ```

2. Click **Lock and Edit.**
3. Navigate to **SecurityRealms**, **<Default Realm Name>**, and **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. Ensure that both the **ObSSOCookie** and **OAM_REMOTE_USER** options are selected under active types.
9. Click **Save** when done.
10. Click **Activate Changes** to propagate the changes.
11. Restart the Administration Server and Managed Servers.

Finally, log in as admin to the WLST console at:

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

Then, run the following command:

```
addOAMSSOProvider(loginuri="/${app.context}/adfAuthentication",logouturi="/oamsso/
logout.html")
```

For example:

```
wls:/offline> connect('weblogic','my_password','t3://ADMINVHN:7001')
Connecting to t3:ADMINVHN:7001 with userid weblogic ...
Successfully connected to Admin Server 'AdminServer' that belongs to domain
'bifoundation_domain'.

wls:/bifoundation_domain/serverConfig>
addOAMSSOProvider(loginuri="/${app.context}/adfAuthentication",logouturi="/oamsso/
logout.html")
```

12.3.6.3 Setting the Order of Providers

Reorder the OAM Identity Asserter, OID Authenticator, and Default Authenticator by ensuring that the control flag for each authenticator is set as follows:

- OAM Identity Asserter: REQUIRED
- OID LDAP Authenticator (or OVD LDAP Authenticator): SUFFICIENT
- Default Authenticator: SUFFICIENT

Then, restart the Administration Server, the Managed Servers, and the Oracle Business Intelligence system components.

12.3.7 Configuring Applications

This section explains how to configure applications and contains the following topics:

- Section 12.3.7.1, "Enabling SSO and Oracle Access Manager for Oracle BI EE"
- Section 12.3.7.2, "Enabling SSO and Oracle Access Manager for BI Publisher"
- Section 12.3.7.3, "Enabling SSO and Oracle Access Manager for Oracle BI Search"
- Section 12.3.7.4, "Enabling SSO and Oracle Access Manager for Oracle RTD"

12.3.7.1 Enabling SSO and Oracle Access Manager for Oracle BI EE

Perform the following steps to enable SSO and Oracle Access Manager for Oracle BI EE:

1. Log in to Fusion Middleware Control.
2. Go to **Business Intelligence**, **coreapplication**, **Security**, and **Single Sign On**.
3. Click **Lock and Edit Configuration**.
4. Select **Enable SSO** and select **Oracle Access Manager** for SSO Provider.
5. Configure the login and logout information for the Oracle BI Presentation Services processes by entering the logon and logoff URLs in the following fields:
   - **The SSO Provider Logon URL:** http://*OAM_host*:*OAM_port*/oamsso/login.html
   - **The SSO Provider Logoff URL:** http://*OAM_host*:*OAM_port*/oamsso/logout.html
6. Click **Apply**.
7. Click **Activate Changes**.
8. Restart all Oracle Business Intelligence system components using opmnctl or Fusion Middleware Control.

12.3.7.2 Enabling SSO and Oracle Access Manager for BI Publisher

Perform the following steps to enable SSO and Oracle Access Manager for BI Publisher:

1. In BI Publisher, go to the **Administration > Security Configuration** page to enable SSO.
2. On the Security Configuration Page, provide the following information in the Single Sign-On section:
   a. Select **Use Single Sign-On**.
   b. For **Single Sign-On Type**, select **Oracle Access Manager**.
   c. For **Single Sign-Off URL**, enter a URL of the following format:

      `http://OAM_host:OAM_port/oamsso/logout.html`

   d. For **User Name Parameter**, specify `OAM_REMOTE_USER`.
3. Click **Apply**.
4. Restart the **bipublisher** application from the Administration Console.

12.3.7.3 Enabling SSO and Oracle Access Manager for Oracle BI Search

Perform the following steps to enable SSO and Oracle Access Manager for Oracle BI Search:

1. Open the BISearchConfig.properties file for editing in the following directory:

   `DOMAIN_HOME/config/fmwconfig/biinstances/coreapplication/`

2. Set the value of BIServerSSOUrl to the following:

   https://bi.mycompany.com/analytics

3. Save and close the file.

12.3.7.4 Enabling SSO and Oracle Access Manager for Oracle RTD

This section provides information about Oracle RTD configuration with Oracle Access Manager.

This section contains the following topic:

- Section 12.3.7.4.1, "Avoiding Problems with Decision Center Logout Redirection"

12.3.7.4.1 Avoiding Problems with Decision Center Logout Redirection When Webgate 10*g* against Oracle Access Manager (OAM) 11*g* is configured as the SSO provider for Oracle Real-Time Decisions Decision Center access, logging out of, then back into Oracle RTD Decision Center prompts users for their user name and password credentials on the re-login. To ensure that this occurs correctly, you must configure the following Oracle RTD Decision Center resources in OAM/WebGate as public (unprotected or anonymous access):

1. Decision Center logout URI /ui/do/logout
2. Decision Center images /ui/images/*

12.4 Backing Up the Identity Management Configuration

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

Perform the following steps to back up the configuration at this point:

1. Back up the web tier using the following steps:

   a. Shut down the instance using `opmnctl`.

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

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

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

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

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

   d. Start the instance using `opmnctl`:

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

2. Back up the Administration Server domain directory. Perform a backup to save the domain configuration. The configuration files all exist under the *ORACLE_*

BASE/admin/*domain_name* directory. Run the following command to create the backup:

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

13 Managing Enterprise Deployments

This chapter describes operations that you can perform after you have configured the Oracle Business Intelligence topology, including monitoring, scaling, and backing up the enterprise deployment.

This chapter contains the following topics:

13.1 About Managing Enterprise Deployments

After configuring the Oracle Business Intelligence enterprise deployment, use the information in this chapter to manage the deployment. This chapter includes topics on typical operations like starting, stopping, monitoring, and patching the deployment

At some point, you might need to expand the topology by scaling it up, or out. See Section 13.4, "Scaling Enterprise Deployments" for information about these tasks.

Back up the topology before and after any configuration changes. See Section 13.6, "Performing Backups and Recoveries for Enterprise Deployments" for information.

This chapter also documents solutions for possible known issues that might occur after you have configured the topology in Section 13.9, "Troubleshooting Enterprise Deployments."

13.2 Starting and Stopping Oracle Business Intelligence

To start Oracle Business Intelligence, you must always start the Managed Servers first, before the system components. In addition, any time that you restart the Managed Servers, you must restart the system components.

For additional information, see "Starting and Stopping Oracle Business Intelligence" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

This section contains the following topics:

- Section 13.2.1, "Starting and Stopping Oracle Business Intelligence Managed Servers"
- Section 13.2.2, "Starting and Stopping Oracle Business Intelligence System Components"

13.2.1 Starting and Stopping Oracle Business Intelligence Managed Servers

Perform the following steps to stop, start, or restart Oracle Business Intelligence Managed Servers:

1. Log in to the Oracle WebLogic Server Administration Console.
2. Expand the **Environment** node in the Domain Structure window.
3. Click **Servers**. The Summary of Servers page is displayed.
4. Select the Oracle Business Intelligence Managed Server that you want to manage (for example, **bi_server1**, **bi_server2**, and so on).
5. Perform one of the following actions:
   - To stop the Managed Server, click **Stop**.
   - To start the Managed Server, click **Start**.
   - To restart the Managed Server, first click **Stop** and wait until the server is completely stopped. Then, select the Managed Server again and click **Start**.

13.2.2 Starting and Stopping Oracle Business Intelligence System Components

Perform the following steps to stop, start, or restart Oracle Business Intelligence system components:

1. Log in to Oracle Enterprise Manager Fusion Middleware Control.
2. Expand the **Business Intelligence** node in the Farm_*domain_name* window.
3. Click **coreapplication**.
4. On the Business Intelligence Overview page, click **Stop**, **Start**, or **Restart**.

13.3 Monitoring Enterprise Deployments

For information on monitoring the Oracle Business Intelligence topology, see "Monitoring Service Levels" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

See also "Diagnosing and Resolving Issues in Oracle Business Intelligence" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition* for information about Oracle Business Intelligence log files, including rotating and managing logs.

13.4 Scaling Enterprise Deployments

You can scale up or scale out the Oracle Business Intelligence enterprise topology, as follows:

- When you scale up the topology, you add additional system components to one of the existing nodes in the enterprise topology.
- When you scale out the topology, you add a new node to the topology with a Managed Server and set of system components.

This section includes the following topics:

- Section 13.4.1, "Scaling Up the Oracle Business Intelligence Topology"
- Section 13.4.2, "Scaling Out the Oracle Business Intelligence Topology"

13.4.1 Scaling Up the Oracle Business Intelligence Topology

This procedure assumes that you already have an enterprise topology that includes two nodes, with a Managed Server and a full set of system components on each node. To scale up the topology, you increase the number of system components that are running on one of the existing nodes.

Note that it is not necessary to run multiple Managed Servers on a given node.

Perform the following steps to scale up the Oracle Business Intelligence enterprise topology:

1. Log in to Fusion Middleware Control.
2. Expand the **Business Intelligence** node in the Farm_*domain_name* window.
3. Click **coreapplication**.
4. Click **Capacity Management**, then click **Scalability**.
5. Click **Lock and Edit Configuration**.
6. Change the number of **BI Servers**, **Presentation Services**, or **JavaHosts** using the arrow keys.
7. Click **Apply**, then click **Activate Changes**.
8. Click **Overview**, then click **Restart**.

13.4.2 Scaling Out the Oracle Business Intelligence Topology

When scaling out the topology, you add a new Managed Server and set of system components to a new node in the topology (APPHOST3). This procedure assumes that you already have an enterprise topology that includes two nodes, with a Managed Server and a full set of system components on each node.

Prerequisites

Before performing the steps in this section, check that you meet these requirements:

- There must be existing nodes that are running Oracle Business Intelligence Managed Servers within the topology.
- The new node (APPHOST3) can access the existing home directories for Oracle WebLogic Server and Oracle Business Intelligence.
- When an *ORACLE_HOME* or *WL_HOME* is shared by multiple servers in different nodes, it is recommended that you keep the Oracle Inventory and Middleware home list in those nodes updated for consistency in the installations and application of patches. To update the oraInventory in a node and "attach" an installation in a shared storage to it, use `*ORACLE_HOME*/oui/bin/attachHome.sh`. To update the Middleware home list to add or remove a *WL_HOME*, edit the *MW_*

HOME/.home file. See the steps in Section 13.4.2.1, "Scale-out Procedure for Oracle Business Intelligence".

- You must ensure that all shared storage directories are available on the new node. Ensure that all shared directories that are listed in Section 4.3.4, "Recommended Directory Locations" are available on all nodes, except for the *ORACLE_INSTANCE* directory and the domain directory for the scaled out Managed Server.

  Also, if you use shared storage for the identity keystore and trust keystore that hold the host name verification certificates, then ensure that the shared storage directory is accessible from the scaled-out node (APPHOST3). If you use local directories for the keystores, then follow the steps in Section 10.3, "Enabling Host Name Verification Certificates for Node Manager" to create and configure a local identity keystore for the scaled-out node.

  For example, mount the following directories:

  – Transaction Log directory

  – JMS Persistence Store

  – Global Cache

  – BI Presentation Catalog

  – BI Repository Publishing directory

  – BI Publisher Catalog

  – BI Publisher Configuration Keystore (certs)

  – *MW_HOME*

13.4.2.1 Scale-out Procedure for Oracle Business Intelligence

Perform the following steps to scale out Oracle Business Intelligence on APPHOST3:

1. On APPHOST3, mount the existing Middleware home, which includes the Oracle Business Intelligence installation and (optionally, if the domain directory for Managed Servers in other nodes resides on shared storage) the domain directory, and ensure that the new node has access to this directory, similar to 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:

   ```
   APPHOST3> cd ORACLE_COMMON_HOME/oui/bin/
   APPHOST3> ./attachHome.sh -jreLoc MW_HOME/jdk
   ```

   To update the Middleware home list, create (or edit, if another WebLogic installation exists in the node) the *MW_HOME*/.home file and add *ORACLE_BASE*/product/fmw to it.

3. Run the Configuration Assistant from one of the shared Oracle homes, using the steps in Section 9.3, "Scaling Out the Oracle Business Intelligence System on APPHOST2" as a guide.

   > **Note:** Before running the Configuration Assistant, ensure that the appropriate JDK version is installed on the same directory as for the first two nodes. If the appropriate JDK is not installed, then the Configuration Assistant likely fails to start.

4. Scale out the system components on APPHOST3, using the steps in Section 9.3.2, "Scaling Out the System Components" as a guide.
5. Configure the bi_server3 Managed Server by setting the Listen Address and disabling host name verification, using the steps in Section 9.4, "Configuring the bi_server2 Managed Server" as a guide.
6. Configure JMS for BI Publisher, as described in Section 9.5.3.1.3, "Configuring JMS for BI Publisher."
7. Set the location of the default persistence store for bi_server 3, as described in Section 9.6.1, "Configuring a Default Persistence Store for Transaction Recovery."
8. Configure Oracle HTTP Server for APPHOST3VHN1, using the steps in Section 8.4.1, "Configuring Oracle HTTP Server for the Administration Server and the bi_servern Managed Servers" as a guide.
9. Start the bi_server3 Managed Server and the system components that are running on APPHOST3. See Section 13.2, "Starting and Stopping Oracle Business Intelligence" for details.
10. Add System Properties to the Server Start Tab for Oracle RTD, as described in Section 9.5.2.2, "Adding System Properties to the Server Start Tab."
11. Configure server migration for the new node, as described in the following sections:
    - Section 11.4, "Enabling Host Name Verification Certificates"
    - Section 11.5, "Editing the Node Manager Properties File"
    - Section 11.6, "Setting Environment and Superuser Privileges for the wlsifconfig.sh Script"
    - Section 11.7, "Configuring Server Migration Targets"
    - Section 11.8, "Testing the Server Migration"
12. To validate the configuration, access the following URLs:
    - Access http://APPHOST3VHN1:9704/analytics to verify the status of bi_server3.
    - Access http://APPHOST3VHN1:9704/wsm-pm to verify the status of Web Services Manager. Click **Validate Policy Manager**. A list of policies and assertion templates that are available in the data is displayed.

      Note: The configuration is incorrect if no policies or assertion templates are displayed.
    - Access http://APPHOST3VHN1:9704/xmlpserver to verify the status of the BI Publisher application.
    - Access http://APPHOST3VHN1:9704/ui to verify the status of the Oracle RTD application.
    - Access http://APPHOST3VHN1:9704/mapviewer to verify the status of MapViewer.
    - Access http://APPHOST3VHN1:9704/aps/Essbase to verify the status of the Oracle Essbase application.
    - Access http://APPHOST3VHN1:9704/aps/SmartView to verify the status of the SmartView application.

- Access http://APPHOST3VHN1:9704/workspace to verify the status of the Workspace application.
- Access http://APPHOST3VHN1:9704/hr to verify the status of the Financial Reporting application.
- Access http://APPHOST3VHN1:9704/calcmgr/index.htm to verify the status of the Calculation Manager application.

13. Oracle recommends using host name verification for the communication between Node Manager and the servers in the domain. This requires the use of certificates for the different addresses that communicate with the Administration Server and other servers. See Chapter 10, "Setting Up Node Manager for an Enterprise Deployment" for further details.

13.5 Manually Failing Over the Administration Server to APPHOST2

In case a node fails, you can fail over the Administration Server to another node. This section describes how to fail over the Administration Server from APPHOST1 to APPHOST2, and contains the following topics:

- Section 13.5.1, "Assumptions and Procedure"
- Section 13.5.2, "Validating Access to APPHOST2 Through Oracle HTTP Server"
- Section 13.5.3, "Failing the Administration Server Back to APPHOST1"

13.5.1 Assumptions and Procedure

Note the following assumptions:

- The Administration Server is configured to listen on ADMINVHN, and not on ANY address.
- The Administration Server is failed over from APPHOST1 to APPHOST2, and the two nodes have these IP addresses:
  - APPHOST1: 100.200.140.165
  - APPHOST2: 100.200.140.205
  - ADMINVHN: 100.200.140.206. This is the VIP where the Administration Server is running, assigned to eth*X*:*Y*, available in APPHOST1 and APPHOST2.
- The domain directory where the administration server is running in APPHOST1 is on a shared storage and is mounted also from APPHOST2.
- Oracle WebLogic Server and Oracle Fusion Middleware components have been installed in APPHOST2 (that is, the same paths for *ORACLE_HOME* and *MW_HOME* that exist on APPHOST1 are also available on APPHOST2).

Procedure

Perform the following steps to fail over the Administration Server to a different node (APPHOST2):

1. Stop the Administration Server, if it is running.
2. Migrate the IP address to the second node using the following steps:
   a. Run the following command as root on APPHOST1 (where *X*:*Y* is the current interface used by ADMINVHN):

```
APPHOST1> /sbin/ifconfig ethX:Y down
```

b. Run the following command on APPHOST2:

```
APPHOST2> /sbin/ifconfig interface:index IP_address netmask netmask
```

For example:

```
APPHOST2> /sbin/ifconfig eth0:1 10.200.140.206 netmask 255.255.255.0
```

Note: Ensure that the netmask and interface to be used match the available network configuration in APPHOST2.

---

c. Update the routing tables using `arping`. For example:

```
APPHOST2> /sbin/arping -b -A -c 3 -I eth0 100.200.140.206
```

3. Start Node Manager in APPHOST2, using the instructions in Step 1 of Section 8.3.3, "Starting the Administration Server on APPHOST1."
4. Start the Administration Server on APPHOST2, using the instructions in Step 2 of Section 8.3.3, "Starting the Administration Server on APPHOST1."
5. Test that you can access the Administration Server on APPHOST2, as follows:
   a. Ensure that you can access the Administration Console at http://ADMINVHN:7001/console.
   b. Check that you can access and verify the status of components in Fusion Middleware Control at http://ADMINVHN:7001/em.

13.5.2 Validating Access to APPHOST2 Through Oracle HTTP Server

Perform the steps that are described in Section 8.4.5, "Validating Access Through Oracle HTTP Server" to check that you can access the Administration Server when it is running on APPHOST2.

13.5.3 Failing the Administration Server Back to APPHOST1

This step checks that you can fail back the Administration Server; that is, stop it on APPHOST2 and run it on APPHOST1 again. Perform the following steps to migrate ADMINVHN back to the APPHOST1 node:

1. Ensure that the Administration Server is not running.
2. Run the following command as root on APPHOST2.

   ```
   APPHOST2> /sbin/ifconfig ethX:Y down
   ```

3. Run the following command on APPHOST1:

   ```
   APPHOST1> /sbin/ifconfig interface:index IP_Address netmask netmask
   ```

   For example:

   ```
   APPHOST1> /sbin/ifconfig eth0:1 100.200.140.206 netmask 255.255.255.0
   ```

Note: Ensure that the netmask and interface that you use matches the available network configuration in APPHOST1.

---

4. Update the routing tables using `arping`. For example:

```
APPHOST1> /sbin/arping -b -A -c 3 -I ethZ 100.200.140.206
```

5. Start the Administration Server again on APPHOST1, as described in Step 2 of Section 8.3.3, "Starting the Administration Server on APPHOST1."
6. Test that you can access the Administration Console at http://ADMINVHN:7001/console.
7. Check that you can access and verify the status of components in Fusion Middleware Control at http://ADMINVHN:7001/em.

If you encounter problems with Administration Server failover, then see Section 13.9.2, "Administration Server Fails to Start After a Manual Failover" for additional information.

13.6 Performing Backups and Recoveries for Enterprise Deployments

See "Backup and Recovery Recommendations for Oracle Business Intelligence" in *Oracle Fusion Middleware Administrator's Guide* for full information about backing up and recovering Oracle Business Intelligence.

13.7 Patching Enterprise Deployments

See "Patching Oracle Business Intelligence Systems" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition* for more information about Oracle Business Intelligence patching.

13.8 Preventing Timeouts for SQLNet Connections

Much of the EDG production deployment involves firewalls. Because database connections are made across firewalls, Oracle recommends that the firewall be configured so that the database connection is not timed out. For Oracle Real Application Clusters (Oracle RAC), the database connections are made on Oracle RAC VIPs and the database listener port. You must configure the firewall to not time out such connections. If such a configuration is not possible, then set the`*SQLNET.EXPIRE_TIME=n*` parameter in the *`ORACLE_HOME`*`/network/admin/sqlnet.ora` file on the database server, where *n* is the time in minutes. Set this value to less than the known value of the timeout for the network device (that is, a firewall). For Oracle RAC, set this parameter in all of the Oracle home directories.

13.9 Troubleshooting Enterprise Deployments

This section includes the following topics:

- Section 13.9.1, "Page Not Found When Accessing BI Applications Through Load Balancer"
- Section 13.9.2, "Administration Server Fails to Start After a Manual Failover"
- Section 13.9.3, "Error While Activating Changes in Administration Console"
- Section 13.9.4, "bi_server Managed Server Not Failed Over After Server Migration"
- Section 13.9.5, "bi_server Managed Server Not Reachable From Browser After Server Migration"
- Section 13.9.6, "OAM Configuration Tool Does Not Remove URLs"

- Section 13.9.7, "Users Redirected to Login Screen After Activating Changes"
- Section 13.9.8, "Users Redirected to Home Page After Activating Changes"
- Section 13.9.9, "Configured JOC Port Already in Use"
- Section 13.9.10, "Out-of-Memory Issues on Managed Servers"
- Section 13.9.11, "Missing JMS Instances on BI Publisher Scheduler Diagnostics Page"
- Section 13.9.12, "BI Publisher Jobs in Inconsistent State After Managed Server Shutdown"
- Section 13.9.13, "JMS Instance Fails In a BI Publisher Scheduler Cluster"
- Section 13.9.14, "Configuring MapViewer when the Administrator Belongs to a Custom Group"

13.9.1 Page Not Found When Accessing BI Applications Through Load Balancer

Problem: A 404 "page not found" message is displayed in the web browser when you try to access Oracle Business Intelligence applications (such as Oracle BI Presentation Services, BI Publisher, and Oracle RTD) using the load balancer address. The error is intermittent and the Managed Servers for Oracle Business Intelligence are shown as "Running" in the Administration Console.

Solution: Even when the Managed Servers for Oracle Business Intelligence are up and running, some of the applications that are contained in them might be in Admin, Prepared, or other states different from Active. The applications might be unavailable while the Managed Server is running. Check the Deployments page in the Administration Console to verify the status of the affected application. It should be in "Active" state. Check the output log for the Managed Server for errors that pertain to that application and try to start it from the Deployments page in the Administration Console.

13.9.2 Administration Server Fails to Start After a Manual Failover

Problem: The Administration Server fails to start after the Administration Server node fails and manual failover to another nodes is performed. The Administration Server output log reports the following:

```
<Feb 19, 2009 3:43:05 AM PST> <Warning> <EmbeddedLDAP> <BEA-171520> <Could not
obtain an exclusive lock for directory: ORACLE_BASE/admin/edg_domain/aserver/edg_
domain/servers/AdminServer/data/ldap/ldapfiles. Waiting for 10 seconds and then
retrying in case existing WebLogic Server is still shutting down.>
```

Solution: When restoring a node after a node crash and using shared storage for the domain directory, you might see this error in the log for the Administration Server due to unsuccessful lock cleanup. To resolve this error, remove the file `ORACLE_BASE/admin/domain_name/aserver/domain_name/servers/AdminServer/data/ldap/ldapfiles/EmbeddedLDAP.lok`.

13.9.3 Error While Activating Changes in Administration Console

Problem: Activation of changes in Administration Console fails after changes to a server's start configuration have been performed. The Administration Console reports the following when clicking "Activate Changes":

```
An error occurred during activation of changes, please see the log for details.
 [Management:141190]The commit phase of the configuration update failed with an
```

```
exception:
In production mode, it's not allowed to set a clear text value to the property:
PasswordEncrypted of ServerStartMBean
```

Solution: This might happen when start parameters are changed for a server in the Administration Console. In this case, provide user name and password information in the server start configuration in the Administration Console for the specific server whose configuration was being changed.

13.9.4 bi_server Managed 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 start. The server seems to be failed over as reported by Node Manager's output information. The VIP used by the bi_server Managed Server is not enabled in the failover node after Node Manager tries to migrate it (if config in the failover node does not report the VIP in any interface). Executing the command "sudo ifconfig $INTERFACE $ADDRESS $NETMASK" does not enable the IP in the failover node.

Solution: The rights and configuration for `sudo` execution do not prompt for a password. Verify the configuration of `sudo` with the system administrator so that `sudo` works without a password prompt.

13.9.5 bi_server Managed Server Not Reachable From Browser After Server Migration

Problem: Server migration is working (bi_server Managed Server is restarted in the failed over node), but the *Virtual_Hostname*:9704/analytics URL cannot be accessed in the web browser. The server has been "killed" on its original host, and Node Manager in the failover node reports that the VIP has been migrated and the server started. The VIP that is used by the Managed Server for bi_server cannot be pinged from the client's node (that is, the node where the browser is being used).

Solution: The `arping` command that is executed by Node Manager to update ARP caches did not broadcast the update properly. In this case, the node is not reachable to external nodes. Either update the nodemanager.properties file to include the MACBroadcast or execute a manual arping:

```
/sbin/arping -b -q -c 3 -A -I INTERFACE ADDRESS > $NullDevice 2>&1
```

where *INTERFACE* is the network interface where the virtual IP is enabled and *ADDRESS* is the virtual IP address.

13.9.6 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 adds new URLs to existing policies only 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 My Policy Domains, and click the created policy domain (bifoundation_domain). Click the **Resources** tab, and then remove the incorrect URLs.

13.9.7 Users Redirected to Login Screen After Activating Changes

Problem: After configuring Oracle HTTP Server and LBR to access the Administration Console, some activation changes cause the redirection to the login screen for the Administration Console.

Solution: This is the result of the console attempting to follow changes to port, channel, and security settings as a user makes these changes. For certain changes, the console might redirect to the Administration Server's listen address. Activation is completed regardless of the redirection. It is not required to log in again; users can simply update the URL to admin.mycompany.com/console/console.portal and directly access the home page for the Administration Console.

> **Note:** This problem does not occur if you have disabled tracking of the changes that are described in this section.

13.9.8 Users Redirected to Home Page After Activating Changes

Problem: After configuring OAM, some activation changes cause redirection to the Administration Console home page (instead of the context menu where the activation was performed).

Solution: This is expected when OAM SSO is configured and the Administration Console is set to follow configuration changes (redirections are performed by the Administration Server when activating some changes). Activations complete regardless of this redirection. For successive changes not to redirect, access the Administration Console, choose Preferences, then Shared Preferences, and deselect the "Follow Configuration Changes" check box.

13.9.9 Configured JOC Port Already in Use

Problem: Attempts to start a Managed Server that uses the Java Object Cache, such as OWSM Managed Servers, fail. The following errors are shown 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.

13.9.10 Out-of-Memory Issues on Managed Servers

Problem: You are experiencing out-of-memory issues on Managed Servers.

Solution: Increase the size of the memory heap that is allocated for the Java Virtual Machine to at least one gigabyte:

1. Log in to the Administration Console.
2. Click **Environment**, then **Servers**.
3. Click a Managed Server name.
4. Open the **Configuration** tab.
5. Open the **Server Start** tab in the second row of tabs.
6. Include the memory parameters in the **Arguments** box. For example:

```
-Xms256m -Xmx1024m -XX:CompileThreshold=8000 -XX:PermSize=128m
-XX:MaxPermSize=1024m
```

> **Note:** The memory parameter requirements might differ between various JVMs (Sun, JRockit, or others).

7. Save the configuration changes.
8. Restart all running Managed Servers.

13.9.11 Missing JMS Instances on BI Publisher Scheduler Diagnostics Page

In some cases, only one JMS instance is visible on the BI Publisher Scheduler diagnostics page, rather than all instances in the cluster. This issue is most likely caused by clocks being out of sync. See Section 2.5, "Clock Synchronization" for more information on the importance of synchronizing clocks on all nodes in the cluster.

13.9.12 BI Publisher Jobs in Inconsistent State After Managed Server Shutdown

Before stopping the Managed Server on which BI Publisher is running, it is a best practice (but not mandatory) to wait for all running BI Publisher jobs to complete, or to cancel any unfinished jobs using the Report Job History page. Otherwise, the shutdown might cause some jobs to incorrectly stay in a running state.

13.9.13 JMS Instance Fails In a BI Publisher Scheduler Cluster

On rare occasions, a JMS instance is missing from a BI Publisher Scheduler cluster. To resolve this issue, restart the BI Publisher application from the Oracle WebLogic Server Administration Console.

Perform the following steps to restart the BI Publisher application:

1. Log in to the Administration Console.
2. Click **Deployments** in the Domain Structure window.
3. Select **bipublisher(11.1.1)**.
4. Click **Stop**.
5. After the application stops, click **Start**.

13.9.14 Configuring MapViewer when the Administrator Belongs to a Custom Group

If the WebLogic domain administration account uses a different group name then Administrators (the default), you must update the MapViewer weblogic.xml file on all nodes to include the actual group name.

Perform the following steps to update the MapViewer weblogic.xml file with the custom group name:

1. Open the MapViewer weblogic.xml file for editing in the following directory:

   ```
   ORACLE_HOME/bifoundation/jee/mapviewer.ear/web.war/WEB-INF
   ```

2. In the weblogic.xml file, locate the following lines:

   ```
   <security-role-assignment>
      <role-name>map_admin_role</role-name>
      <principal-name>Administrators</principal-name>
   ```

```
</security-role-assignment>

<security-role-assignment>
   <role-name>secure_maps_role</role-name>
   <principal-name>Administrators</principal-name>
</security-role-assignment>
```

3. Replace the two occurrences of Administrators with the actual administration group name (for example, BIAdministrators). For example:

   ```
   <security-role-assignment>
      <role-name>map_admin_role</role-name>
      <principal-name>BIAdministrators</principal-name>
   </security-role-assignment>

   <security-role-assignment>
   <role-name>secure_maps_role</role-name>
   <principal-name>BIAdministrators</principal-name>
   </security-role-assignment>
   ```

4. Save and close the file.

5. Restart the MapViewer application, as follows:

   a. Log in to the Administration Console.

   b. Click **Deployments** in the Domain Structure window.

   c. Select **mapviewer(11.1.1)**.

   d. Click **Stop**.

   e. After the application has stopped, click **Start**.

6. Repeat these steps on all nodes.

Index